Investigation and Application of Digital Signal Processing and Wavelet Technologies to Automatic Coin Recognition

Darren James Sharman

A submission presented in partial fulfilment of the
requirements of the University of Glamorgan/Prifysgol Morgannwg
for the degree of Master of Philosophy

This Research program was carried out
in collaboration with Tetrel Technology Ltd.

May 1999

To my Father,

for engaging my interest in engineering,

and then giving me the encouragement I needed

to pursue it as a career.

To my Mother,

for her endless patience,

and for always believing in me.

I'm very proud of them both.

Abstract

This thesis examines the application of Digital Signal Processing (DSP) techniques, and specifically Wavelets, to the field of automatic coin recognition. The aim is to utilise DSP techniques to exploit information that is contained within time domain signals representing coins, which can not be accessed by other means. Attention is also given to the power requirement of possible solutions, with a low power solution being a secondary aim, as the solutions are targeted for use in a line-powered payphone.

An examination of existing coin recognition techniques is presented, for which an improved but basic DSP coin recognition scheme using peak and trough location, is achieved. This is then improved using more advanced DSP techniques to access previously unavailable information contained within the signals.

The advanced DSP techniques are developed into an integrated framework for automatic coin recognition. The framework is used to identify a single Wavelet solution that supplies a DSP representation of a set of coins. The representations of different coin types exist within a region of n-dimensional Euclidean space, which the framework attempts to locate uniquely for each coin type.

To enable the framework to operate successfully, a key feature presented is the resampling of the waveforms input into the framework, to normalise any temporal variations in the input data.

The location of the single Wavelet for analysis can not be achieved analytically and so is obtained using a novel Data Mining solution to search a Wavelet dictionary for possible solutions. This thesis proves that utilisation of the time localisation properties of the Discrete Wavelet Transform is possible when taken together with a distance metric strategy.

Appropriate results are presented to verify the performance of the Wavelet solutions provided by the framework, especially in respect of counteracting fraudulent coins in the recognition process. As an overall validation of the research solution, an emulation of the coin recognition system was produced that could validate coins in real time, this is also documented.

Both the hardware and software components of the integrated framework which have been developed, are fully modular and hold significant potential for expansion and integration into newer, more powerful cost effective coin recognition systems.

Acknowledgements

I would like to thank the following people:

Dr. Laurence Dooley – for his assistance throughout this research project, without whose help this research would not have reached completion.

Mr Jim Churchman – for initiating this project and for his continued inspiration, support and technical excellence throughout its execution

Mr Jeff Harris and Mr Tony Walytschko – for their assistance in all matters electronic

Mr Kevin Maidment – for turning me into a Visual C++ programmer against all the odds, and putting up with 1001+ questions

Mr Ceri Peart – for the great ideas about using state machines in my microprocessor software

Mr David Cottrell – for helping me to understand the requirements that a completed coin mechanism would have to meet both in performance and design for manufacture

Mr Ian Corp – for his assistance with PCB layout of the prototypes used in this project

Mr Andrew Jones – for his assistance with mechanical drawings in this thesis

Mr Chris Griffths – for all matters secretarial

My thanks also go to all the members of staff at Tetrel Technology not already mentioned above, The University of Glamorgan and the Teaching Company Scheme who have contributed to the successful completion of this research project.

Lastly, my thanks go to Miss Simone Rosenfielder for her continued patience during my long hours in front of the computer, and for contributing her voice to the *WaveActiveX* real time coin mechanism simulation.

Glossary of Terms (Nomenclature)

ASIC	Application Specific Integrated Circuit
CWT	Continuous Wavelet Transform
DTW	Dynamic Time Warping
DWT	Discrete Wavelet Transform
EEPROM	Electronically Erasable Programmable Read Only Memory
EPROM	Erasable Programmable Read Only Memory
FFT	Fast Fourier Transform
FIFO	First In First Out
FWT	Fast Wavelet Transform
MAF	Moving Average Filter
Opto	Optical Transistor
PCB	Printed Circuit Board
QMF	Quadrature Mirror Filters
STFT	Short Time Fourier Transform

Mathematical Conventions

t	continuous time
n or k	discrete time
N, N_0	the set of natural numbers $N_0 = N \cup \{0\}$
R	the set of real numbers
Z	the set of integers
Δt	A small change in time
Δf	A small change in frequency
s	signal or waveform
ω_0	resonant operating frequency
Q	a measure of the efficiency of an inductor
R(i)	autocorrelation value at i shifts of the multiplier waveform
X(f)	Short Time Fourier Transform
C(j,k)	Discrete Wavelet Transform
C(a,b)	Continuous Wavelet Transform
ψ	Wavelet Function
g_k	Discrete Wavelet ($k \in Z$)
$a = 2^{-j}, j \in Z$	Dyadic scale, 2^{-j} is the resolution or j is the level
$\frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right)$	Family associated with the one-dimensional Wavelet $a > 0$, $b \in R$

Contents

Table of Figures

Chapter 1 - Introduction

This research was initiated by Tetrel Technology as part of a Teaching Company Scheme, in a response to a requirement to improve their existing range of low-cost payphone coin recognition systems.

The company already had an existing coin mechanism in production, which was offering the minimal functionality that was required to recognise coins. The coin mechanism had several limitations however, which needed to be overcome in order to retain Tetrel competitiveness in what remains a highly volatile market place.

The limitations included the facts that the existing coin mechanism could only recognise a maximum set of six coins, it was also susceptible to fraud using metallic disks of similar size and shape to the coins which were in the coin mechanism's coin recognition set. The coin mechanism was also difficult to manufacture reliably and it required a great deal of human intervention to program with the required coin recognition set.

The objective of this project therefore, was to seek a viable solution that would overcome these problems by providing a larger coin set, decreased possibility of fraud, and also be easily manufactured.

Many options were explored for improving the coin mechanism but these were narrowed both in terms of the intended implementation, which was to be in a standard Tetrel payphone, and in terms of commercial interests. These requirements served to focus the aims and objectives of the research during the initial stages.

The company also wanted a coin mechanism that was very low cost, in the region of £5.00p for the total component cost. This immediately steered away from a solution that involved complex electronic and mechanical components and towards applying certain elementary Digital Signal Processing (DSP) technologies. Clearly, a novel DSP algorithmic implementation, which was capable of utilising coin waveform details, was required. An algorithm that could be highly discriminatory to digital signals representing a coin as it passes through the mechanism.

Various techniques were experimented with, from simple cross correlation of signals continuing through to the eventual solution based upon exploiting Wavelet technologies.

Other design requirements focused on placement of the new coin mechanism into a Tetrel product. Size of the solution was very important as Tetrel already had a range of plastic moulded cases developed for their payphones. Due to the high cost of developing plastics, the new coin mechanism would have to be size compatible with the old.

The final prerequisite placed on the design was current consumption, Tetrel Technology payphones are, in most models, Telephone line powered. This has the effect that only 4mA is available to run the entire coin mechanism on these models.

Applying DSP technologies to overcome the limitations of the existing coin validator was thus seen as a natural evolution for this research. The coin profile as it passes through the mechanism can be digitised and stored. Subtle variations in the characteristic can be extracted in order to assist in either verifying or rejecting the coin in real time. This thesis discusses all aspects of the DSP solution proposed in the context of the various requirements discussed above. It is structured as follows: -

- Chapter 2 discusses the preliminary research that was undertaken and examines previous coin mechanism technologies, and then comments on what was applicable within the development of this research.

- Chapter 3 explores the background theory of Digital Signal Processing techniques that were employed in the development of an algorithm suitable to the task of coin recognition. Particular attention is given to the research progression that resulted in the utilisation of the Wavelet techniques.

- Chapter 4 elucidates the systems implemented to research and exploit the DSP and distance metric techniques, and includes and examination of the hardware and software implementations that were produced.

- Chapter 5 contains details of the methods followed for carrying out analysis using the systems developed in the previous chapter.

- Chapter 6 contains a discussion of the results of the data mining, and presents the evidence that the Wavelet Transform is a viable method of signal analysis for coin recognition.

- Chapter 7 presents the conclusions that can be drawn from this thesis. It also contains recommendations for future work and gives an orientation of how this research will be implemented and advanced.

Chapter 2 - Coin Validator Literature Search

Two useful sources of information were available to the author on coin validator technology, these came from existing Tetrel payphones, and patents belonging to the company. Other sources of information, such as patents for other companies, proved either unhelpful or inaccessible, this was due to the highly competitive market in which the coin mechanisms are sold, and so companies treat such information as 'commercial in confidence' secrets.

Various coin mechanism were available for study and these could be divided into two categories, mechanical and electronic solutions.

2.1 Mechanical Coin Validators

The mechanical solutions consisted of a set of different sized coin entry slots that would only fit certain coin sizes. Further inside the mechanism holes were cut into the coin chutes which would allow coins smaller than the required coin to drop out of the mechanism and not register as a valid coin. Valid coins would proceed down the relevant chute and make an electrical connection between contacts spaced at a valid coin diameter apart to register a valid coin entry. This method was very primitive and left the mechanism open to wide fraudulent abuse to triggering by any metallic objects which were similar in size and shape to the target coins. Even simple attempts at coin fraud were successful, for example using a metallic washer. The simplicity with which these types of coin mechanisms could be deceived prevents them from producing a robust solution when used alone. Nevertheless, the principle of exploiting the size attributes of coins was considered as a possible method of pre-processing prior to any DSP solution.

2.2 Electronic Coin Validators

The electronic solutions all are based around the same fundamental principle, the effect that a metallic object has on an Electro-magnetic field that forms part of an oscillating circuit. This principle is exploited in two forms, either using capacitance (Churchman 1997, see appendix A) or inductance (Tetrel 1997,see appendix B) as the transducer mechanism.

The capacative method involves passing the coin between two large plates that are forming a capacitor, which is part of an oscillating circuit. When the coin comes between the plates, it in effect forms two capacitors in series (as shown in Figure 2.1). This in turn modifies the frequency of the oscillator circuit of which the capacitor forms part. The amount by which the frequency changes is dependent on both the thickness and diameter of the coin which is in the field. The capacative oscillator circuit is operated at approximately 4Mhz, this is an arbitrary value chosen as a trade off between a lack of sensitivity at lower frequencies, and the need for special considerations for the design of the circuit at higher frequencies. Examples of the special considerations which would be required are, supply decoupling, stray coupling within the circuit, high frequency Printed Circuit Board (PCB) track layout difficulties.

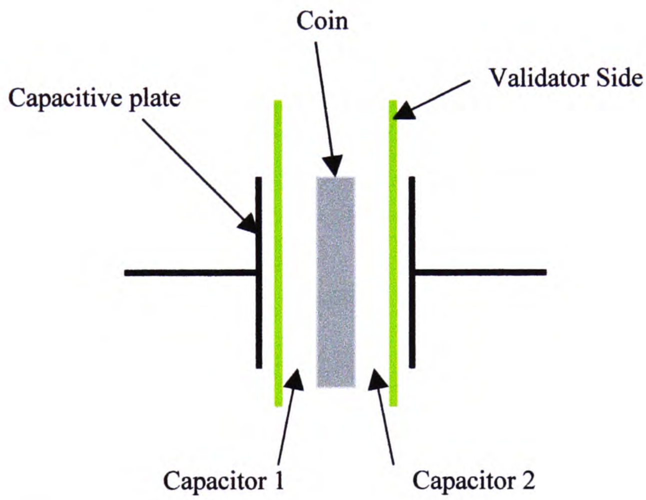


Figure 2.1 – The Effect of a Coin on the Capacitve Transducer

The inductive method involves passing a coin near to an inductor, which again is forming part of an oscillating circuit. The inductor is positioned such that the direction of coin travel is perpendicular to the magnetic field generated by the inductor. This allows maximal flux cutting to occur when the coin passes. When the coin enters the flux of the inductor the self-inductance, that the field around the inductor is generating, is disturbed in two ways. Firstly, the coin has the effect of shortening the magnetic flux path of the inductor. Secondly, there is conduction of current within the coin setting up eddy currents that incur losses within the magnetic field. Both of these effects will change the inductance of the inductor, which then alters the operating frequency of the oscillator. The oscillator circuit is operated at around 8Khz as this provides information about the metallic properties of the coin and specifically the ferromagnetic composition of the coin. In practice, operating frequencies between 5kHz – 20kHz provide adequate results dependant on the coin types tested.

Another method, which can be used to alter the operating frequency of the oscillator, is instead of monitoring inductance changes, to monitor the losses incurred by the flux passing through the coin. The losses manifest themselves as a change in the 'Q' (see Equation 2.1) which is a measure of the efficiency of the inductor. The overall effect on the oscillator is the same with a given characteristic change in frequency being observed for different coins. The coin mechanism circuits that were developed for experimental use in this research all utilised the inductance monitoring method.

$$Q = \frac{\omega_0 L}{R}$$ **Equation 2.1**

where :

Q – Energy stored per cycle ÷ energy lost per cycle (coulombs) [this is only true at resonant frequency]
ω_0 – resonant operating frequency
L – inductance (henrys)
R – combination of losses (ohms)

The capacative and inductive transducers are combined in two possible ways to produce a signal which is representative of the coin passing through the mechanism. The combination used depends on the metallic properties of the coin set to be recognised, and which configuration will produce signals that are sufficiently different to allow coin recognition to occur. The two implementation options are to either have a capacative transducer and a fixed period low frequency oscillator, or to have a capacative transducer with an inductive Transducer.

Both configurations operate in similar ways, but the inductive and capacative coin mechanism possess the most similarity to the coin mechanisms which were developed for use in this research project. For this reason, the inductive and capacative transducer configuration will be used to illustrate the circuit operation of the coin mechanism. Figure 2.2 represents a simplified view of the coin mechanism circuit developed by Tetrel Technology, with a full circuit diagram being available in Appendix C.

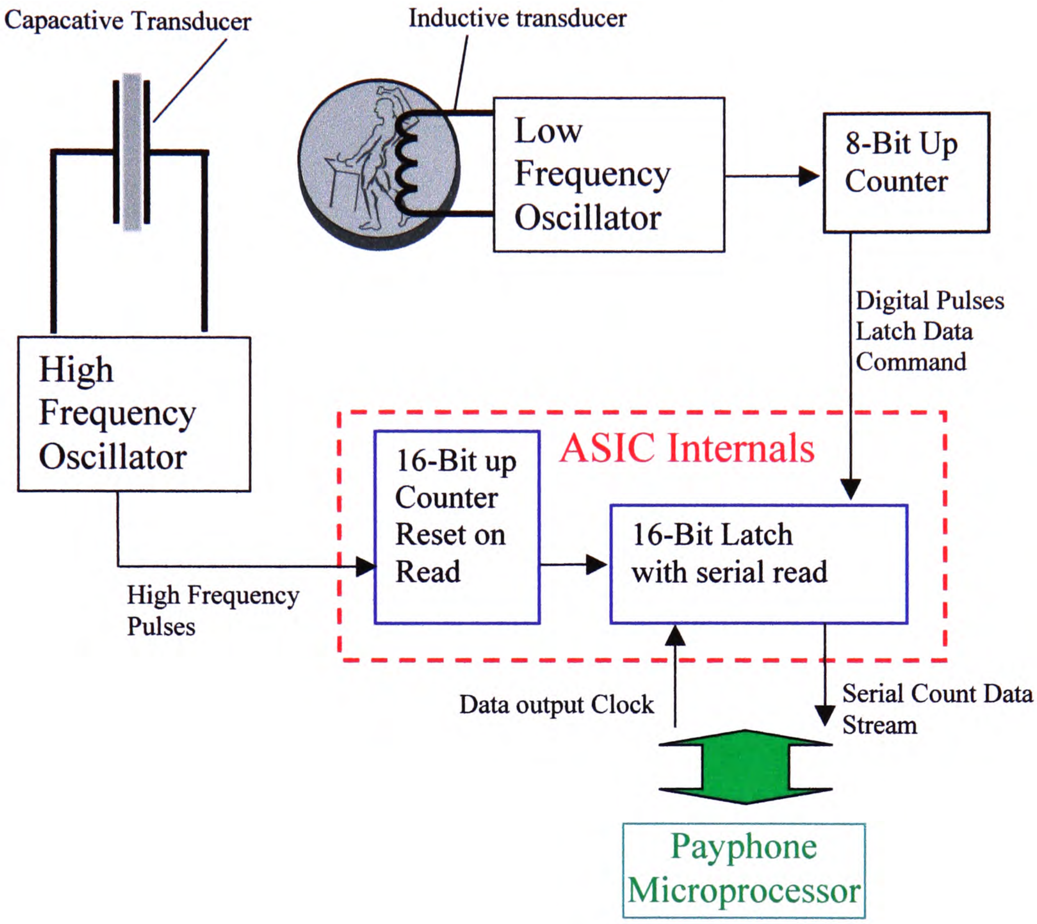


Figure 2.2 – Tetrel Technology Inductive and capacative coin mechanism block circuit diagram

The main features of the circuit are as follows:

High Frequency Oscillator – Generates pulses at a frequency that can be modified by the capacative transducer

Low Frequency Oscillator – Generates pulses at a frequency that can be modified by the inductive transducer

16-Bit Counter – Counts high frequency pulses and resets to zero when read by the latch

8-Bit Counter – Counts low frequency oscillator pulses, generates a pulse when required count is reached then resets

16-Bit Latch – Latches 16-Bit Counter reading after a pulse from the 8-Bit counter is received. In addition, the latch places data onto the serial data line, one bit at a time, for transmission to the payphone processor when data out clock pulses are received.

When the coin mechanism is operating, an Application Specific Integrated Circuit (AISC) is used to count the high frequency pulses for the period of one low frequency oscillator cycle. When a low frequency pulse is received the high frequency counts are stored then shifted out onto a serial data line.

The serial data line is received by a microprocessor that is controlling all of the payphone functions. The processor applies an algorithm that is used to find peaks and troughs in the signal. The peaks and troughs are then used to characterise the coins and to decide if the coin is valid or invalid. The limits, within which the peaks and troughs must fall, are pre-programmed for all the valid coins in the data set. An Example of a coin waveform, which would be contained in the data stream, is shown in Figure 2.3 the diagram also shows an example of the coin limits for this coin.

The electronic solutions have superseded the original mechanical implementations used in all applications in Tetrel Technology's products and offer a considerably better protection against fraud than the mechanical option. The electronic method was ideal for upgrading into a DSP scheme as the origins of the data were already in the digital domain in the form of pulses from the oscillators. In addition, Digital Signal Processing would be able to exploit information from within the actual coin waveforms not simply information about the amplitudes of the peaks and troughs.

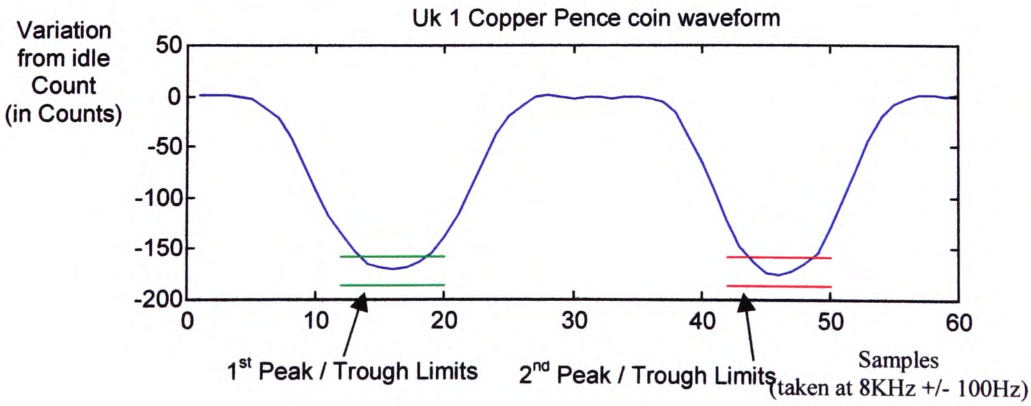


Figure 2.3 – Example of coin data from serial data stream and coin limits

Chapter 3 - Data Processing and Decomposition Methods

This chapter presents a discussion of the digital signal processing techniques and the development of distance metrics that were fundamental to the research undertaken.

A progression from simpler DSP techniques, such as Autocorrelation, towards the eventual solution of Wavelets can be observed within the text. The chapter is presented in this way, as this was an important philosophy behind the research. It was stated in the introduction that a potential solution should be low cost and this suggests utilising a simple DSP algorithm requiring the lowest possible processing power budget. It therefore follows that the research progressed from the simpler algorithms to the more complex until a suitable solution was found.

To enable the information extracted by the DSP techniques to be used in making recognition decisions, a method of examining the resultant data was required. The distance metrics examined towards the end of this chapter fulfil this requirement.

3.1 Autocorrelation

A good starting point for the research was to use autocorrelation, this gives a numerical representation of the frequency content of a signal (Proakis, 1996, p. 118-133). The equation for autocorrelation is shown in Equation 3.1

$$R(i) = \sum_{n=1}^{N-i} s(n)s(n-i)$$

Equation 3.1

where:

R(i) – autocorrelation value at index and shift position i
n – current waveform sample
N – total waveform length
i – autocorrelation number and shift position

Consider Figure 3.1 and Figure 3.2, these graphs show the autocorrelation functions of two samplings of coin waveforms from two sample coins. Each waveform has been autocorrelated with itself (not its similar partner waveform from the same coin, otherwise this would be correlation) and the four resulting autocorrelation functions have been plotted below the respective coin waveforms used to generate them. The waveforms in Figure 3.1 contain information that is of a higher frequency than those of Figure 3.2. This is reflected in the autocorrelation of the signals, because each frequency component in the original waveform gives rise to a term in the autocorrelation function and Figure 3.1 has a more complex autocorrelation function.

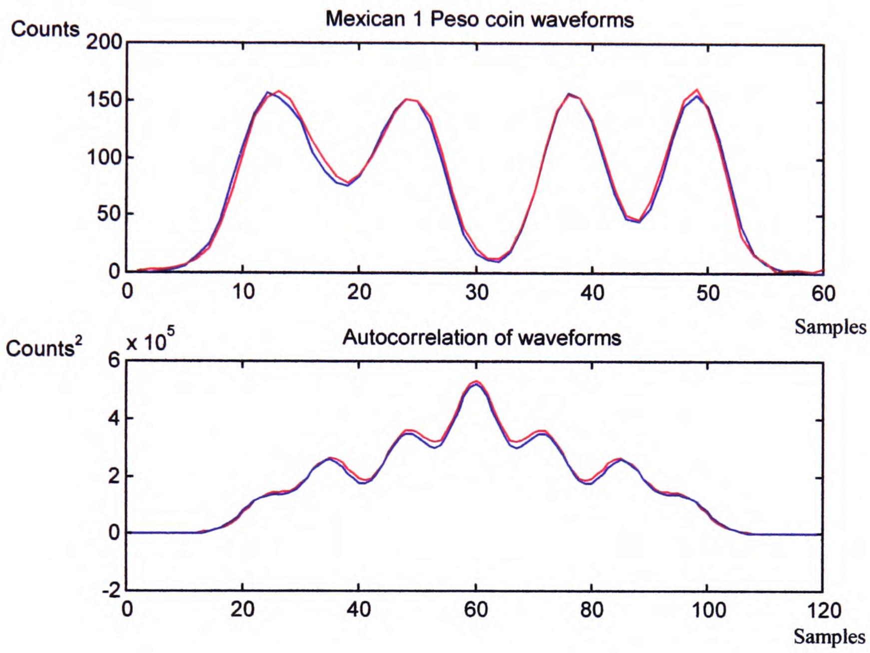


Figure 3.1 – Autocorrelation of two samples of Mexican 1 peso coin waveforms

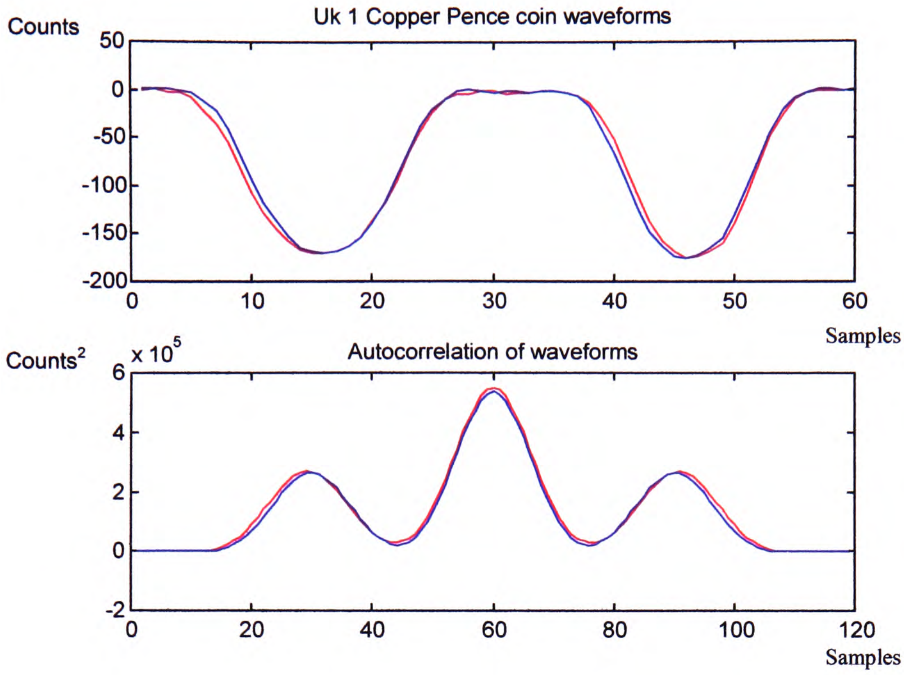


Figure 3.2 – Autocorrelation of two samples of UK 1 Copper Pence coin waveforms

The main problem with using autocorrelation for recognition purposes is the way in which the frequencies of a waveform are represented after the autocorrelation has taken place. In Figure 3.3 an autocorrelation of a coin waveform with a discontinuity is shown. This discontinuity has been deliberately added to demonstrate that information that can point to where the detail, which caused the high frequency components is in the waveform, is not available from the autocorrelation function.

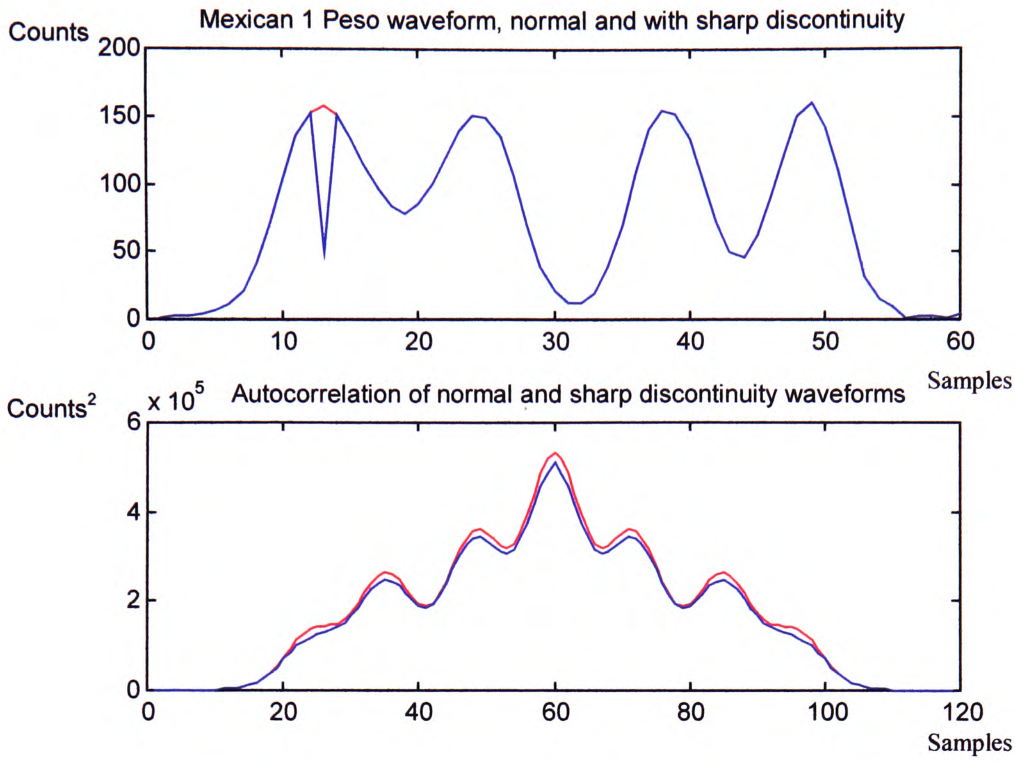


Figure 3.3 – Autocorrelation of Mexican 1 Peso waveform with and without sharp discontinuity added

3.2 Matched Filtering

Matched filtering is a technique that involves matching the response of a filter to a desired input waveform. It is similar in principle to a child's toy, where the child must match different shaped pegs into the right shaped holes. The response of a filter is tuned such that any waveform entering will produce low output unless it exactly matches the tuned filter response. Figure 3.4 elucidates this concept, the figure shows waveform data entering a filter, the centre section of which is matched to the filter. The response of the filter can be seen to be zero (in practice the value is almost zero) until the waveform exactly matches the required response.

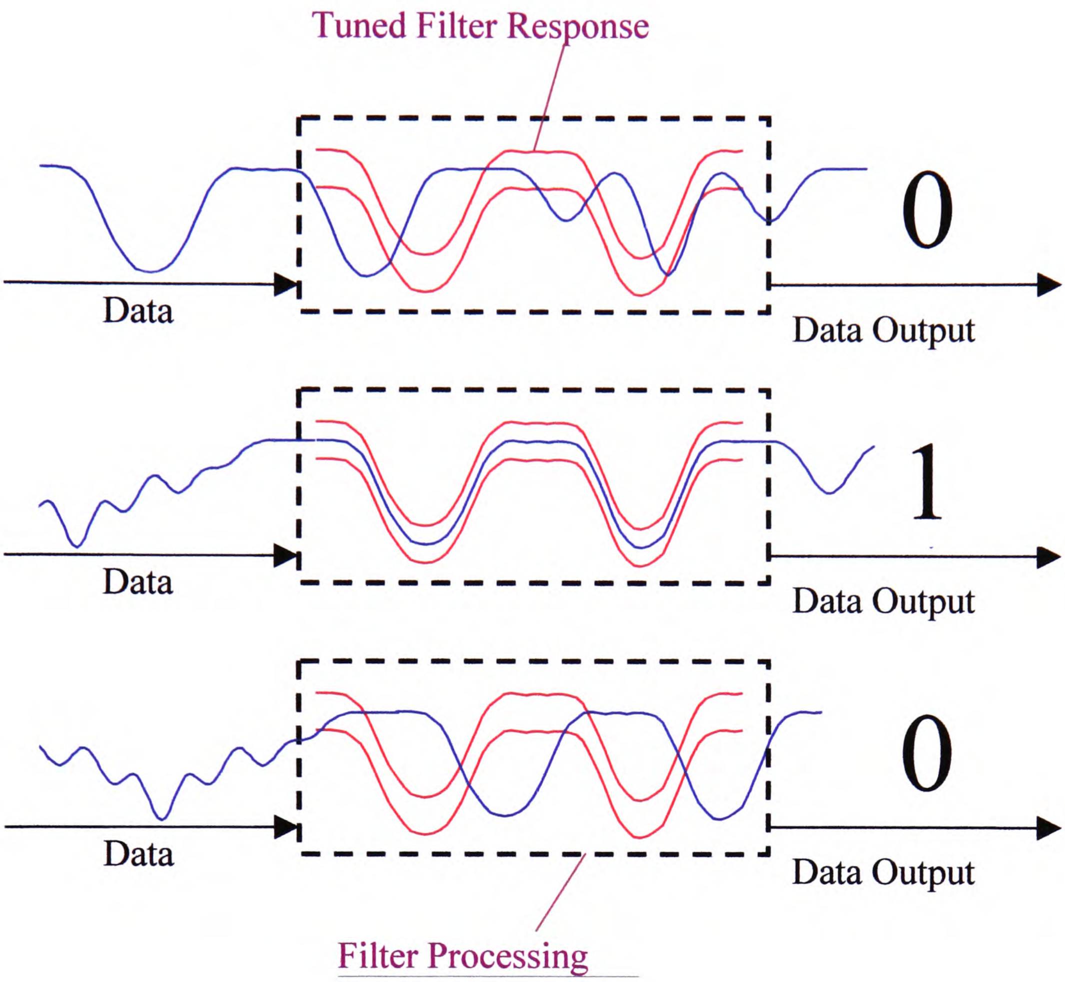


Figure 3.4 – Matched Filter processing Example

On first appraisal, this technique can appear to be ideal for the task of coin recognition, with the data processing and recognition actions being performed simply within the filter. When the holistic view of the coin recognition task that a coin validation system is expected to perform is taken, then this is not the case.

Consider that for each coin in the recognition set a separate filter would have to be designed, this would result in an inflexibility when new coins were required to be introduced to the coin mechanism recognition set. In addition, when a system of this type would be operated, a coin waveform would have to be processed by each filter, which would be large processing task. Lastly, the method of mechanical handling used by the coin mechanism, on which the new recognition scheme is to be based, would prove unworkable with this type of system. This is because the coins can roll through the mechanism at different speeds, resulting in expanded or contracted versions of the coin waveforms. Matched filters are not structured to deal with these variations which can occur within the coin waveforms.

3.3 *The Fourier Transform*

The Fourier transform decomposes the target waveform into its composite frequency components. Hubbard (1996, p. 96 - 103) states, it uses sines and cosines, which oscillate indefinitely, for the analysing function. There is a fast method in existence which is known as the Fast Fourier Transform (FFT) which takes $n \log_2 n$ computations to compute the transformation of a signal with n points. The infinite nature of the analysing functions makes this method suitable for repetitive signals that are completely predictable over a period.

This technique was used initially to analyse the coin waveforms to understand their nature. Transforming the coin waveforms into the frequency domain yielded poor results for similar reasons as those experienced with Autocorrelation. The coin waveforms were of such a low frequency that the Fourier Transform could not deliver much information about the content of the signals. In addition, when the transform decomposed the coin waveforms into their component sinusoids, it was realised that the waveforms were made up of many sinusoids but that they were contained in a very small low frequency band. Again, as with autocorrelation, the transformed data contained no time localised information as to where the main details of the coin waveforms were occurring. This served to smear any fast, transient signal details across the whole transform, effectively masking and preventing their use for separating some coin types.

3.4 *Short Time Fourier Transform*

The Short Time Fourier Transform (STFT) was invented in 1946 by Dennis Gabor, this technique modifies the operation of the Fourier transform to operate only on a short section of the target waveform at a time. Akansu (1992, p.292) describes it as using a wave limited within a time window, multiplied by trigonometric oscillations to perform the analysis. The size of the window is fixed for each analysis but the frequencies inside the window vary. This modified transformation offers interesting properties for control of the Fourier analysis. If the window function is made small more accurate time information is contained within the Transformation, this unfortunately has the compromise that low frequency information is lost. Conversely if the window is made large, information about low frequency signals is available but at the cost of precision time information. This situation of compromise makes this transform suited to quasi-stationary signals, which are considered stationary at the size of the window.

The sectioning of the waveform is achieved by placing a mathematical window around a region of data, outside of which the waveform is considered to have zero amplitude. Equation 3.2 shows the STFT, it positions the window $\omega(n_0)$ at some point n on the time axis and calculates the Fourier Transform of the signal within the extent or spread of the window

$$X(f) = \int_{-\infty}^{\infty} [x(n)\omega(n_0 - n)]e^{-j2\pi fn} dn$$

Equation 3.2

Where:

$X(f)$	- is the STFT of the waveform
$x(n)$	- is the original waveform
$\omega(n_0)$	- is the window function
n	- is the index position of the window

The effect of the STFT is demonstrated in Figure 3.5, in effect it can be seen as a succession of Fourier Transforms of a windowed segment of the signal (vertical stripes) or as a modulated analysis filter bank (horizontal stripes).

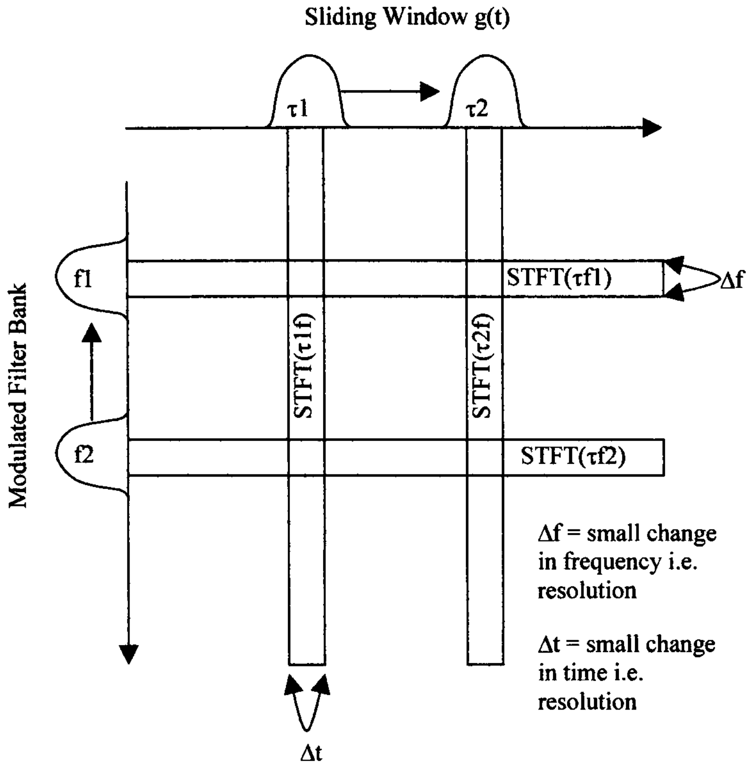


Figure 3.5 – The Time-Frequency plane corresponding to the Short-Time Fourier Transform.

The main problem with the application of this technique for feature extraction from the coin waveforms is that the selection of window size is critical in what type of results that can be obtained. Once the window length is selected then both Δt and Δf (the time and frequency resolutions) are fixed and cannot be changed. This means that the realisation of either good time or good frequency resolution cannot be achieved at the expense of the other parameter, as shown in Figure 3.6.

Clearly, there are many applications where this restriction is unacceptable and an improved method would be one that affords a multiresolution solution. This involves selecting a window depending on the input coin signal features so that time and frequency resolution can be exchanged to provide either small Δf and large Δt or vice versa.

It is for this reason that Wavelets have become such a popular signal processing technique and seem ideal for exploitation in this application. Wavelets will now be examined in the following section.

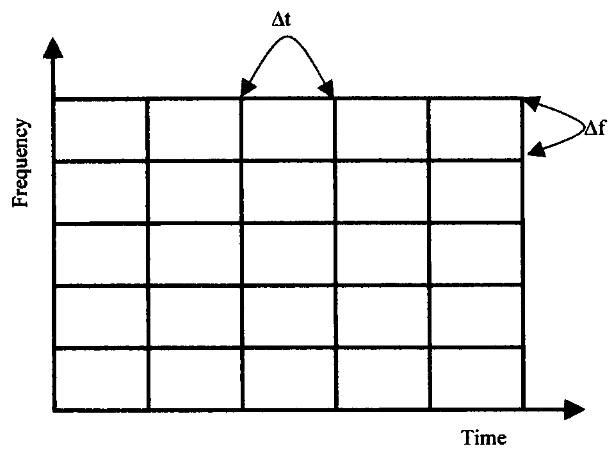


Figure 3.6 – The effect that a fixed window size has in the time and frequency domains

3.5 The Wavelet Transform

Figure 3.7 illustrates how Wavelets compares against the other methods of signal decomposition discussed in this chapter. The axis labels for the Wavelet diagram are scale and time, Daubechies (1992) states this as the essential feature of the Wavelet Transform.

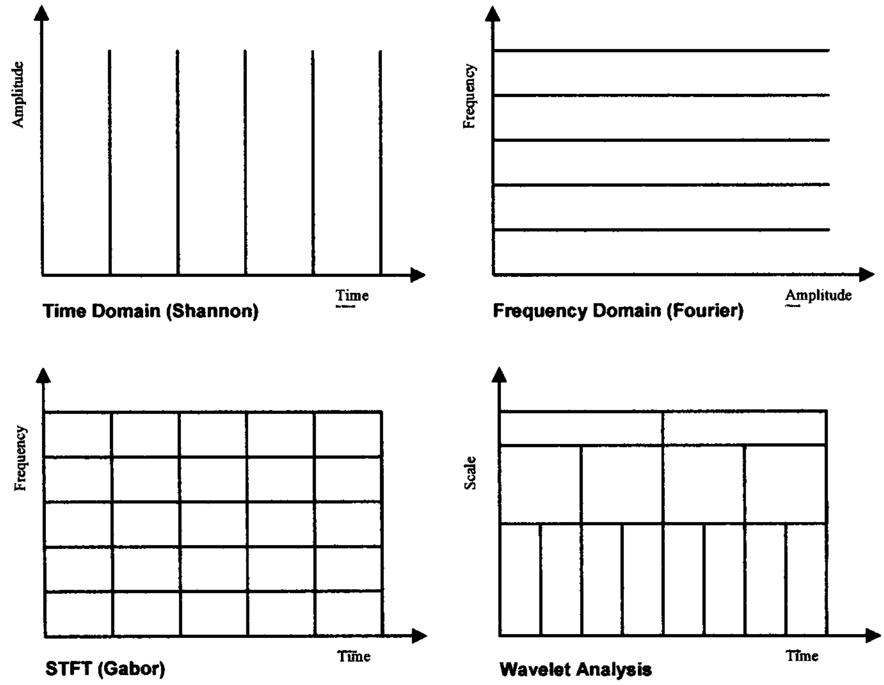


Figure 3.7 – Contrasting Signal Domains (Hubbard, 1996)

The Wavelet transform provides an interesting form of decomposition that is time-scale based. 'The analysing function is a signal limited in time with a fixed number of oscillations which integrates to zero, this is the Wavelet from which the technique derives its name. During the analysis, the Wavelet forms the window with which the target signal is compared, and it is dilated and contracted to change its size. Throughout the dilations and contractions, or scale changes, the Wavelet retains the same number of oscillations and so the frequency to which the Wavelet is sensitive changes. Small Wavelets provide localised time information but are insensitive to low frequency information. Large Wavelets provide good frequency information but provide poor time localisation. The advantage of this technique over the windowed Fourier transform is that all of this information is available simultaneously. The

Wavelet transform is suited both for analysing transient or very short time signals, and as mentioned earlier, the time localisation properties can be very advantageously exploited in trying to discriminate subtle signal variations in the coin profiles. Hubbard (1996) states that the discrete orthogonal form of the transform is also very computationally attractive, with only CN computations required, where C is the complexity of the Wavelet and M is the number of points in the analysed signal. When this is compared against the N $\log_2$ N of the Fourier Transform, the advantages of the Wavelet Transform can easily be seen.

3.5.1 The Continuous Wavelet Transform

The Continuous Wavelet Transform (CWT), shown in Equation 3.3, is defined as the sum overall time of the signal multiplied by scaled, shifted versions of the Wavelet function ψ (Matlab, 1998).

(N.B. the entire mathematical notation used below is covered in the mathematical conventions at the front of this thesis)

$$C(a,b) = \int s(t)\frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right)dt$$ **Equation 3.3**

Given: $a \in \Re^{+} - \{0\}$ and $b \in \Re$

Where:

a	– scale (or dilation) of the Wavelet
b	– translation (or time position) of the Wavelet
s(t)	– is the input signal
C(a,b)	– is s(t) decomposed into the CWT coefficients
ψ	- is the mother Wavelet function

Akansu (1992, p. 310) states that this transformation contains information but in theory, will be infinitely redundant and computationally impossible to implement as the Wavelet can be shifted to and infinite number of positions and dilated to an infinite number of scales.

3.5.2 The Discrete Wavelet Transform

When using the discrete Wavelet transform, a Wavelet is translated and dilated only by discrete values. The discrete values chosen for the dilation are mostly indicated as powers of two (sometimes known as dyadic scales), this is because these scales yield completely orthogonal transforms. The word Orthogonal, in this case means 'without redundancy'.

Converting the CWT formula into DWT yields Equation 3.4.

$$C(a,b) = C(j,k) = \sum_{n \in Z} s(n) g_{j,k}(n)$$

Equation 3.4

Given: $a = 2^j$, $b = k2^j$, $j \in \mathrm{N}$ and $k \in \mathrm{Z}$

Where:

j	– scale (or dilation) of the Wavelet
k	– translation (or time position) of the Wavelet
s(n)	– is the input signal
C(j,k)	– is s(t) decomposed into the DWT coefficients
$g_{j,k}$	- is the Wavelet filter

As can be seen from Equation 3.4 the Wavelet Ψ has been replaced by the Wavelet filter $g_{j,k}$ this is defined by Equation 3.5.

$$g_{j,k}(n) = 2^{\frac{-j}{2}} g(2^{-j} n - k)$$

Equation 3.5

3.5.3 The Fast Wavelet Transform

Due to the demands placed on the coin mechanism to make recognition decisions quickly, and also the need for a solution to run on a low power processor, the DWT needed to be optimised. The Fast Wavelet Transform (FWT) provides such an optimisation.

Ironically, the fastest way of computing the Wavelet Transform does not involve using Wavelets at all (Hubbard, 1996, p. 152). Mallat (1989, p. 674-693) developed the fast implementation from multiresolution theory, it operates using a series of cascaded filters.

The Mallat algorithm for the Fast Wavelet Transform (FWT) is a classical scheme in the signal processing community, known as a two-channel subband coder using conjugate quadrature filters or quadrature mirror filters (QMF). The two filters used in implementing this scheme for the FWT have the properties that the impulse responses of the filters are the Wavelet and its complementary Scaling function.

A basic overview of one stage of the FWT operation can be seen in Figure 3.8. In the diagram, the FWT can be seen to be utilising a decimation section, decimation is simply the act of removing samples from the waveform. In this case the decimation is by a factor of two which means removing every second sample.

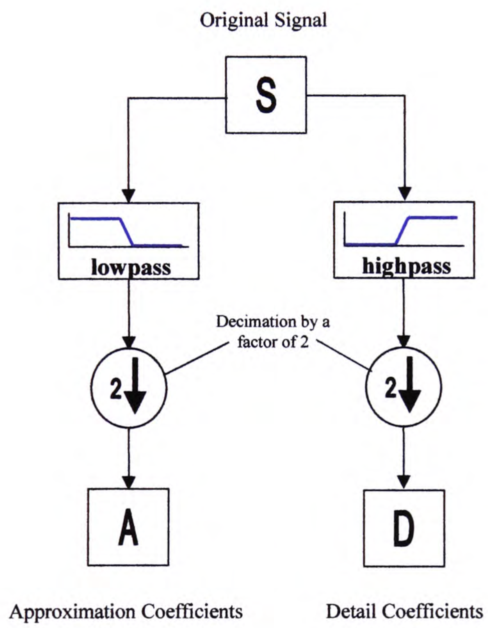


Figure 3.8 – One Stage of the Fast Wavelet Transform

The high-pass filter is associated with the Wavelet and passes only high frequency signal components that are associated with the variations or details of the signal. The low-pass filter is associated with the Scaling function (a function which is the starting point for orthogonal decompositions) which smoothes the details out of the remaining data not encoded by the Wavelet.

The decimation process is used to dispose of redundant samples. The information that was contained in the original sample sequence has now been split between the approximation and detail coefficients. The information is sufficiently halved between the two resultant sets of coefficients such that a decimation by a factor of two is possible. For a mathematical treatment of this process see Louis (1997, p.122).

From Figure 3.8 it can be seen that when the signal is passed once through this arrangement, it emerges in two data streams possessing half the samples of the original. To conduct a second level of decomposition the resultant approximation coefficients are passed back thorough the same filters for a second time, see Figure 3.9. This is what makes the transformation fast as, due to the action of the decimation, now only half as many calculations have to be conducted.

Also highly important to the cost effective coin recognition system being developed, only two sets of filter coefficients have to be stored and can be reused as for each decomposition that is required.

The feeding back of the signal into the filters can be continued until all the detail information has been 'drained' out of the original signal and encoded into Wavelet coefficients. This forms a signal decomposition tree. Interestingly the FWT is fully invertable, no information is lost, so in using this transformation there is a guarantee of obtaining all the information from the coin waveforms for use in recognition.

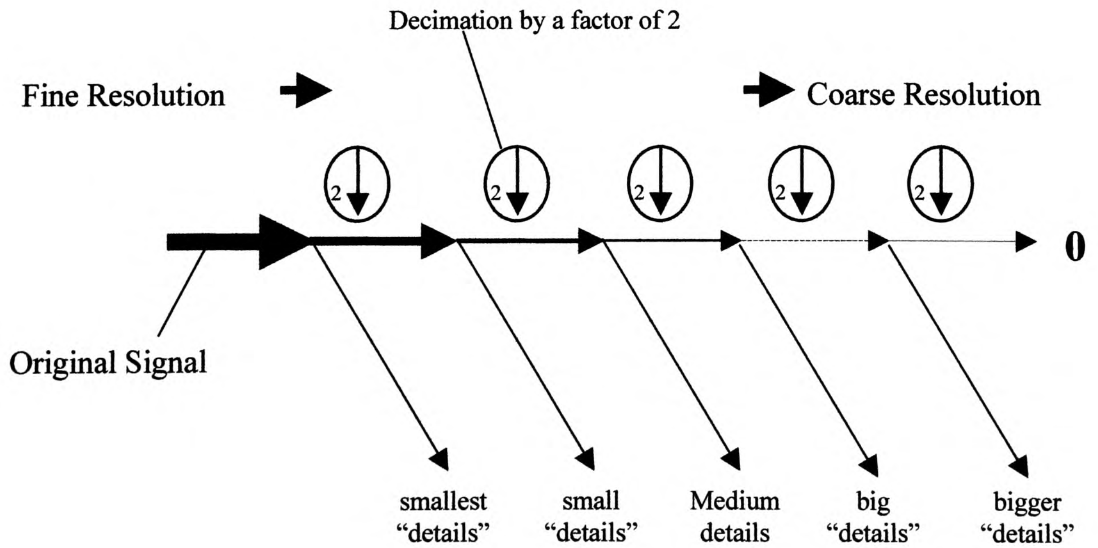


Figure 3.9 – The action of the Fast Wavelet Transform

This subsection has give a cursory overview of the underlying principles and theory of Wavelets, a much more comprehensive analysis of these techniques can be found in the following references Daubechies (1992), Daubechies (1990), Daubechies (1988), Wickerhauser (1994), Mac (1992), Meyer (1993), Meyer (1991), Akansu (1992), Louis (1997), Torrance (1997), Kaiser (1994), Mulcahy (1992), Streng (1993) Ruskai (1991) and Chui (1992).

3.5.4 Wavelet Packets

The Wavelet packet transform decomposes a signal into time-frequency scale components and can be useful when analysing signals that are composed of stationary and non-stationary components. The analysing function looks like a Wavelet multiplied by a trigonometric function.

This method offers no additional information in its decompositions than that which is available from Wavelet Transform when applied to the coin signals. This is due to the very short time and low frequency nature of the waveforms. This transform could only serve to increase the computational load of any solution that involved Wavelets in the context of the coin waveforms. The reader can pursue these further with Daubechies (1992, p. 331) and Mac (1992).

3.6 Wavelet Selection – Data Mining

In considering the use of Wavelets for recognition, a critical choice is which type of Wavelet is to be used for the analysis. In this section the novel method for locating a single Wavelet, which is the 'best fit' for recognition of a set coins, is developed.

3.6.1 Background Theory

Using Wavelets for feature extraction from coin waveforms for use in recognition has an intrinsic difficulty. The technique offers an 'embarrassment of riches' (Hubbard, 1996, p. 194) in possible implementations, which prevented following a direct analytical approach to an ideal Wavelet solution. For example, the Wavelet transform can be implemented with an almost infinite number of different Wavelets, but the question is which Wavelet will give the best results? The problem becomes apparent when directly attempting to ascertain which parameters of a Wavelet that extract the best features, arbitrarily, from each coin waveform. A similar problem was tackled by Mallat (1993) with Matching Pursuits. The Matching method can be used to optimise the decomposition of a signal into a linear expansion of waveforms. It effectively searches a redundant dictionary of Wavelet packets to find the optimal path through a Packet Decomposition Tree for a particular type of Wavelet. This paper was expanded on by Liu (1998) in the field of machinery diagnosis, for the purpose of indicating incipient changes in signals relating to machine failure. Instead of using a best matching criteria, which was the main feature of Mallat (1993), Liu uses an information measure to search a redundant Wavelet dictionary in order to locate a small set of Wavelets that are informative for machine failure.

Liu presents an interesting method but this is not suitable for the coin recognition task. In the introduction to this thesis, it was stated that a successful solution would have to be able to be implemented on a low power DSP system. This means that the solution would not be able to have the luxury of using multiple Wavelets to recognise the coins as the computational overhead would be excessive. To counter this, a solution would have to be found that would locate a single Wavelet which would be able to separate a set of coins whilst retaining a localised readings for several samples of the same coin.

It now becomes apparent that an analytical solution to this problem would be excessively complex and so a different philosophy behind the search for the ideal DSP solution was required in the form of Data Mining.

3.6.2 Data Mining

Data Mining basically involves collecting vast amounts of data and processing it in many possible ways then analysing the results for the solution which can best satisfy a set of requirements. As mentioned above, each DSP technique offered an almost infinite set of possibilities, these where narrowed analytically to guide the data mining by giving a smaller set of possible solutions which could be practicably managed.

As an example of the analytical methods involved, consider the implementation of the Wavelet transform. There were many Wavelets easily available for decomposing the coin data. By visual inspection of the Wavelets, some could be easily discounted from further investigation simply by considering how the shape of the Wavelet would correlate with the coin waveforms. In this way the size of the task required of the data mining programs could be kept to a practical level.

The requirements for a successful result from the Data Mining activities that would find a solution suitable for coin recognition, were as follows: -

1. To be able to provide a distance metric between a set of input coin data samples of different coins.

2. To coalesce a set of coin samples from the same coin into a given definable vector space which is separable in at least one dimension from all other coin samples.

3. To provide consistent and repeatable results

Another requirement which can be added to the three above to make the solution appropriate for incorporation into a commercially viable payphone for use by Tetrel Technology is as follows: -

4. To supply a solution which is the most cost-effective in terms requiring the minimum computational effort for reliable recognition to occur.

3.6.3 Wavelet Effects

In this sub-section, a selection of three Wavelets is examined to give the reader a feel for the action that different Wavelet shapes can have. The analysis presented in this section has importance, as this was the method used to prepare the redundant dictionary of Wavelets that is searched by the Data Mining method.

Each Wavelet is considered with an explanation of the action that can be expected when they are applied to coin waveforms.

There are no formulas presented to describe the Wavelets in question, this is because "Wavelets are creatures of the computer age" (Hubbard, 1996, p. 46), they cannot be constructed from analytic formulas, they are produced using iterations of functions.

Wavelets with the rounded curves such as a Daubechies 4 (see Figure 3.10), can be used to push the main curve shapes of coin waveforms into the detail side of the DWT early in the decomposition results tree. While the noisier aspects of the waveform would remain in the approximation coefficients until later into the decompositions.

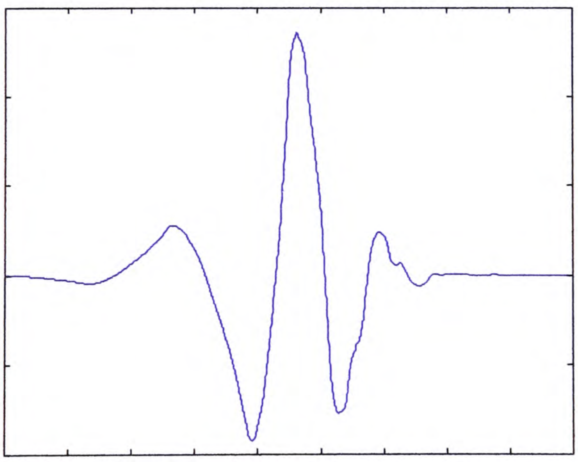


Figure 3.10 – The Daubechies 4 Wavelet

The Wavelets with high frequency content, such as the Coiflets 1 (see Figure 3.11), have the opposite effect to the rounded Wavelets. The transient discontinuities of the data are pushed into the detail side of the transformation results earlier in the decomposition tree, and the low frequency basic waveform shapes are retained in the approximation side of the transform results tree until later into the decompositions.

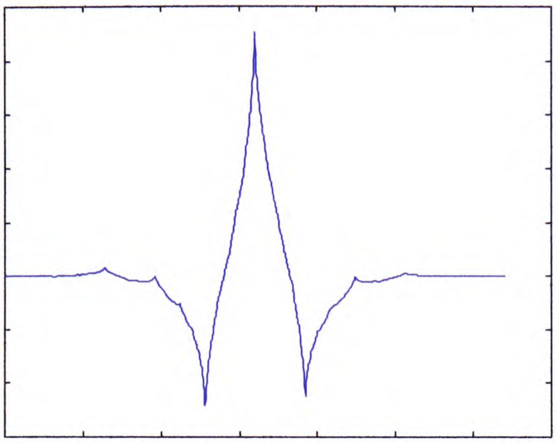


Figure 3.11 – The Coiflets 1 Wavelet

The low computational effort Wavelets can be illustrated by the Haar Wavelet (see Figure 3.12). This has the properties that it is simply a single cycle of a square wave, with amplitudes 1 and –1. The implementation of this Wavelet in software is the simplest of all the Wavelets, which makes for simple filters with results produced quickly.

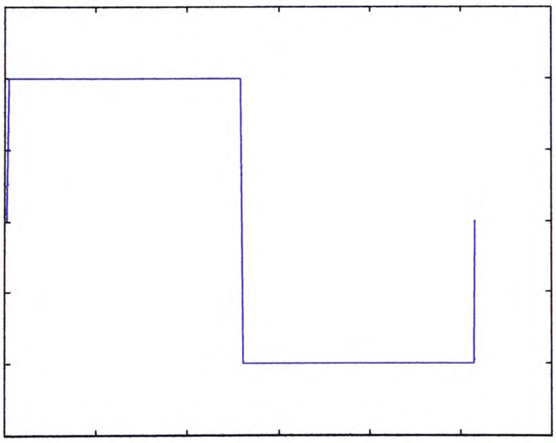


Figure 3.12 – The Haar Wavelet (also known as the Daubechies 1)

From this simple examination the reader can start to appreciate the wealth and diversity of information that Wavelets can extract from data. The question now becomes not of how to discover the information contained within waveforms, but rather of what to do with it when it is discovered? To answer this question the next section presents an examination of distance metrics.

3.7 Development of Distance Metrics

This section covers the development of the distance metrics, which are required to process, and make decisions about, the coefficients which are produced from data processing and decomposition methods. The distance metrics are best understood when placed in context with a task, for this reason the distance metrics are discussed in relation to their implementations in coin recognition when utilised with Wavelets. Although, it should be noted that these metrics can be adapted and utilised with the results obtained from any of the data processing methods discussed in Chapter 3 .

Three distance metric methods were tested, Hyper-spheric, Hyper-cubic and Hyper-elliptic, each increasing in computational effort required and complexity.

3.7.1 Hyper-spheric

This approach used a least squared vector distance to separate the coins.

The DWT coefficient results, obtained from Equation 3.4, can be considered to reside in n-dimensional function space and each coin type can be represented by a hyper-sphere, which contained all coin readings pertaining to each type.

The co-ordinates of the centre of the hyper-sphere for each coin can be calculated using a Trimmed mean of the DWT coefficients in each dimension. A trimmed mean is simply a mean that excludes a percentile of the upper and lower readings of a data set. The radius of the hyper-sphere, representing each coin, is then found from the coin sample that is the largest mean squared distance away from the centre. This is a straight-line magnitude vector, which if it is allowed to rotate through all other dimensions, and to stay a uniform length, defines a hyper-spheric area around the centre point. As the coin reading is the largest distance away from the centre for all the coefficients in any dimension, the effect is to encompass all the other coin readings. To illustrate this point, consider Figure 3.13 that demonstrates how this would be achieved in two dimensions.

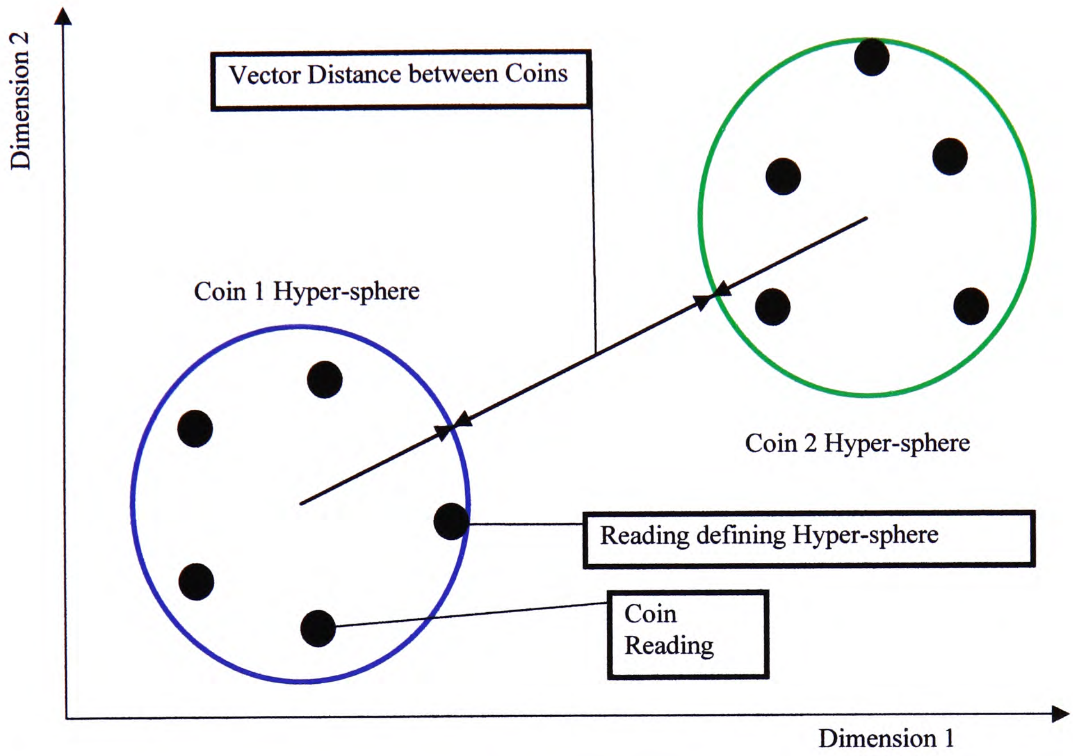


Figure 3.13 – Illustration of Using Least squared distance between coins in 2-D

Once all the hyper-spheres have been calculated for each coin, the task is then to check if any overlap exists. This is calculated for each decomposition level individually, checking two coins at a time and calculating the length of the vector between the hyper-sphere centres. The absolute value of the combined radii of both coins is then subtracted from this centre distance. If the result is negative then the hyper-spheres overlap and the coins cannot be separated. An example calculation for this operation in two dimensions is shown in Equation 3.6 and Equation 3.7.

$$CentDist = \sqrt{(Coin1Dim1 - Coin2Dim1)^2 + (Coin1Dim2 - Coin2Dim2)^2}$$ **Equation 3.6**

$$TotalDist = CentDist - \sqrt{(HyperRadCoin1)^2} - \sqrt{(HyperRadCoin2)^2}$$ **Equation 3.7**

This procedure is then repeated until the distance from each coin to all other coins has been established. A simulation of this operation is shown in Table 3.1, using the approximation results from a Daubechies1 Wavelet at decomposition 10, the red cells in the table show where overlapping hyper-spheres would exist (the coin name abbreviations used in the table can be found in Appendix E). As can be seen from the table, this Wavelet at decomposition level 10 can not be used for recognition as overlapping hyper-spheres exist. If a result for a decomposition level exhibited no overlapping hyper-spheres then the combination of filter and decomposition level would be suitable for coin recognition.

	M_1_Pe	M_2_Pe	M_10_Ne_Pe	M_10_Ol_Pe	M_20_Pe	M_5_Pe	M_50_Ce	U_Qu_Do	U_1_Co_Pe	U_1_Po	U_1_St_Pe	U_10_Pe	U_2_Co_Pe	U_2_Pe	U_2_Po	U_20_Pe	U_5_Pe	U_50_Ne_Pe	U_50_Ol_Pe
M_1_Pe																			
M_2_Pe	-31.55																		
M_10_Ne_Pe	126.6125	121.1188																	
M_10_Ol_Pe	168.2188	162.725	27.96875																
M_20_Pe	172.8188	167.325	32.56875	-14.525															
M_5_Pe	-12.6688	-30.925	140	181.6063	186.2063														
M_50_Ce	120.65	115.1563	-14.4813	27.125	31.725	134.0375													
U_Qu_Do	198.9875	193.4938	58.7375	11.64375	-13.7625	212.375	57.89375												
U_1_Co_Pe	157.6438	152.15	17.39375	-29.7	-32.8375	171.0313	16.55	-6.66875											
U_1_Po	139.3125	133.8188	-0.9375	15.00625	19.60625	152.7	-1.78125	45.775	4.43125										
U_1_St_Pe	-30.2375	-35.7313	99.58125	141.1875	145.7875	-16.85	93.61875	171.9563	130.6125	112.2813									
U_10_Pe	104.5313	99.0375	8.8	50.40625	55.00625	117.9188	2.8375	81.175	39.83125	21.5	77.5								
U_2_Co_Pe	208.2	202.7063	67.95	20.85625	-4.55	221.5875	67.10625	-8.39375	2.54375	54.9875	181.1688	90.3875							
U_2_Pe	207.4375	201.9438	67.1875	20.09375	-5.3125	220.825	66.34375	-9.15625	1.78125	54.225	180.4063	89.625	-26.1875						
U_2_Po	118.2625	112.7688	-14.4813	27.125	31.725	131.65	-20.4438	57.89375	16.55	-1.78125	91.23125	0.45	67.10625	66.34375					
U_20_Pe	107.8438	102.35	0.5875	42.19375	46.79375	121.2313	-5.375	72.9625	31.61875	13.2875	80.8125	-9.96875	82.175	81.4125	-7.7625				
U_5_Pe	96.61875	91.125	19.425	61.03125	65.63125	110.0063	13.4625	91.8	50.45625	32.125	69.5875	-2.65625	101.0125	100.25	11.075	0.65625			
U_50_Ne_Pe	106.175	100.6813	-0.45	41.15625	45.75625	119.5625	-6.4125	71.925	30.58125	12.25	79.14375	-11.6375	81.1375	80.375	-8.8	-19.2188	-1.0125		
U_50_Ol_Pe	118	112.5063	-9.09375	32.5125	37.1125	131.3875	-15.0563	63.28125	21.9375	3.60625	90.96875	0.1875	72.49375	71.73125	-17.4438	-8.025	10.8125	-9.0625	
U_2_St_Pe	-25.5875	-35.35	127.0813	168.6875	173.2875	-16.4688	121.1188	199.4563	158.1125	139.7813	-29.7688	105	208.6688	207.9063	118.7313	108.3125	97.0875	106.6438	118.4688

Table 3.1 -An Example of distances calculated between hyper-spheres

3.7.2 Hyper-elliptics

There is an inherent problem with the Hyper-spheric distance metric is that the distance between coins is derived from a single vector. This does not offer much opportunity to develop distance between the coins and could result in many coin spheres overlapping. The rotation of the vector defining the hyper-sphere could be considered to be too clumsy. In an attempt to overcome this effect weightings can be applied to the vector, this has the effect of changing the spherical shape generated into an ellipsoid. For example, observe the effect on the circle in Figure 3.14 when the coefficients in the y dimension are weighted by a half.

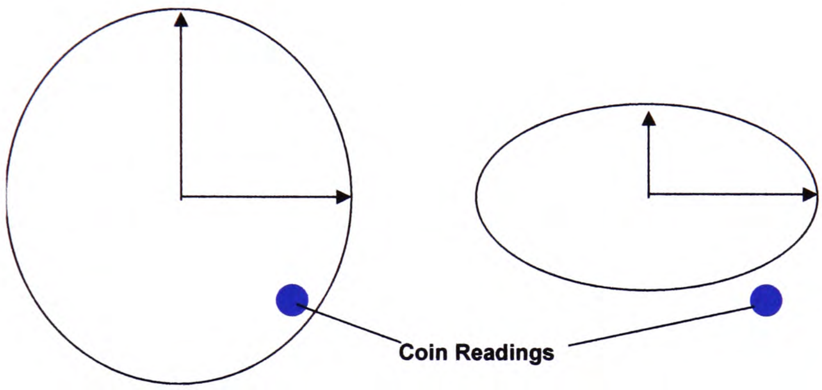


Figure 3.14 – The effect of weighting on spherical vector spaces

Simulations of this technique proved the weighting caused more problems than they solved, as coin readings which were initially encompassed by the spheres were being excluded by the elliptics as shown in Figure 3.14. This coupled with the complexity in finding and implementing valid weighting values made using elliptics less viable than a hyper-cubic solution. Due to these difficulties, elliptics were not used in the data mining investigations discussed in Section 3.6.

3.7.3 Hyper-cubes

This method offers the best opportunity of creating distance between different coin types as it treats each dimension, in the function space of the coefficient results, individually.

When using this method, the coefficient results are considered to reside in n-dimensional function space and each coin type has a hyper-cubic area within the n-dimensional space assigned to it. The centre of each side of the hyper-cube, representing a coin, can be calculated using the Trimmed mean of the coefficients for a coin type. This is carried out at each decomposition level, in each orthogonal dimension. The length of each side of the cube in each dimension is governed by the coin reading that is the furthest away from the centre point. Again, this procedure is illustrated in 2-dimentions in Figure 3.15.

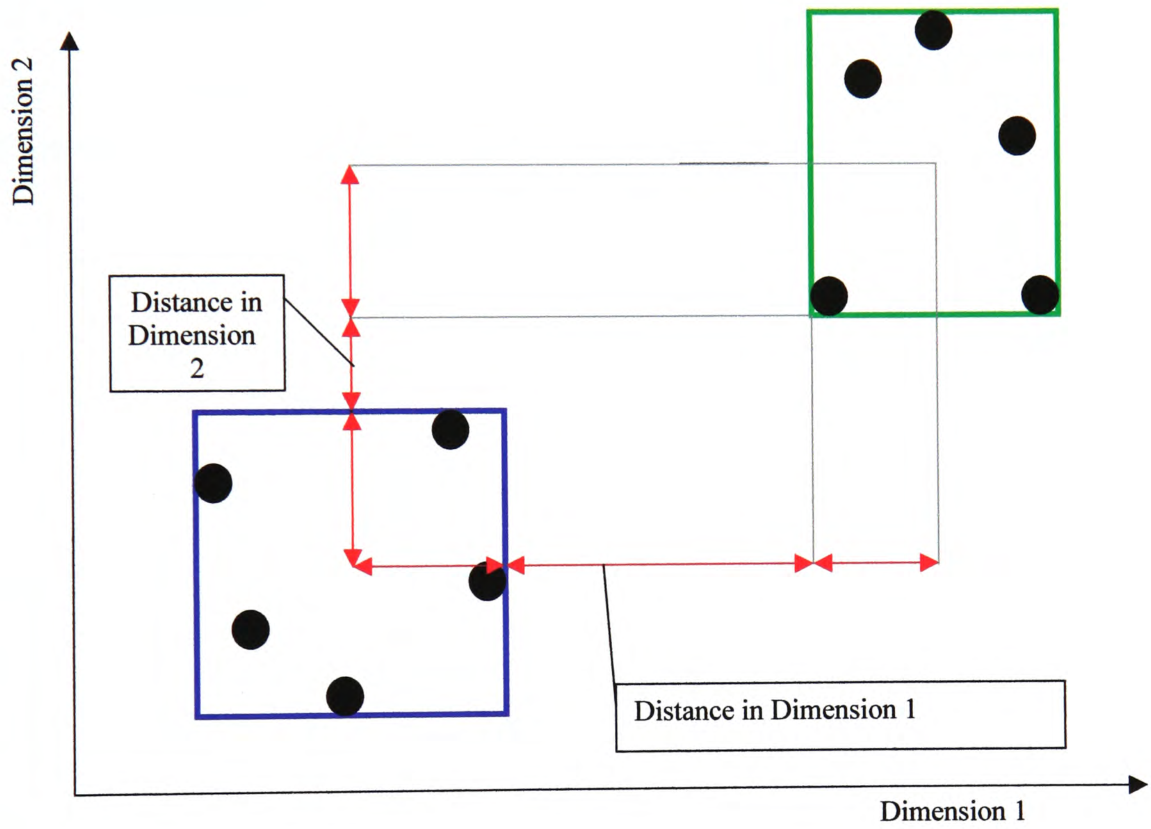


Figure 3.15 – Finding cubic coin limits in 2-D

Checking the resultant coin limits is a large task. Each coin type has a set of upper (UU) and lower (LL) coin limits in each dimension as show in Figure 3.16

Figure 3.16 – Coin limits in Hyper-cubes

In the diagram Coin 3 and 4 can be seen to overlap, so this dimension cannot be used to separate these coins. However, Coins 1 and 2 do not overlap so this dimension can be used to separate these coins.

It now becomes apparent that the task is to compare the limits of each coin against all the other coins in each dimension. To qualify as an applicable recognition solution, each coin must have at least one dimension where it does not clash with the other coins in the recognition set.

This is a very difficult idea to picture, so consider the following simplified two-dimensional simulation shown in Figure 3.17

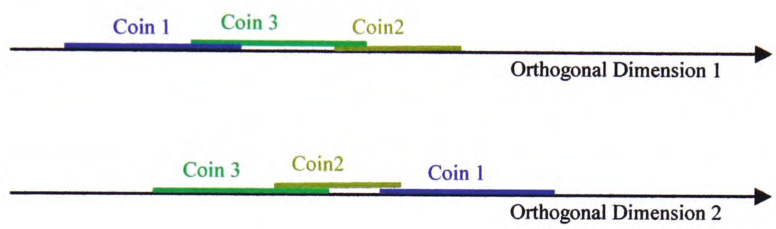


Figure 3.17 – Differing coins clashing in two dimensions

Coin 1 can be separated by these two dimensions whereas coins 2 and 3 cannot. When observing coin 1, it can be seen that it clashes with other coins in both dimensions, the important point being that they are different coin clashes. If, in a recognition scenario, valid values for coin 1 were received in the positions clashing with coins 2 and 3 as shown in Figure 3.18, then coin 1 can be identified as follows.

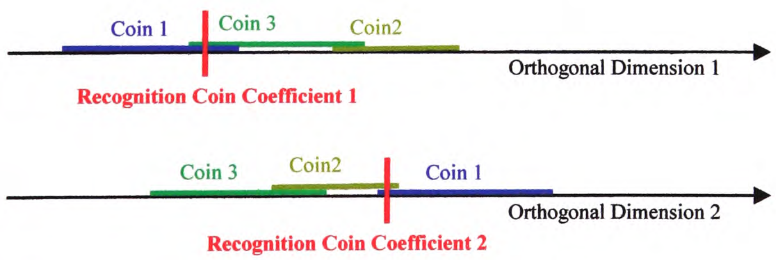


Figure 3.18 – Coin readings in clashing areas

If a bit result array is produced for this task, where '1' is a reading inside a coins limits and '0' is a reading outside a coins limits, the result would be as shown in Table 3.2.

	Dimension 1 Limit Matches	Dimension 2 Limit Matches
Coin 1	1	1
Coin 2	0	1
Coin 3	1	0

Table 3.2 – Example bit result array

From the table we can see that only coin 1 matches the readings in both dimensions and so would be correctly identified.

This algorithm was the basis of the coin recognition strategy that was adopted, though in practice N dimensions rather than the above example of two dimensions were involved.

3.7.4 Distance Metric Development Observations

From the discussion above it can be seen that the hyper-spheric distance metric will offer the most computationally effective solution as each coin would only have to have the co-ordinates of the centre of the hyper-sphere stored and a vector which would dictate a valid coin sphere. It does however, have the drawback that it offers the least opportunity to create distance between the coin types.

The hyper-cubic metric requires more information to be able to separate the coins than the hyper-spheric method, with an upper and lower limit needing to be stored for each dimension of each coin. In a scenario where very precise coin recognition is required, such as when a coin mechanism has a large coin set or where two coins are very similar and could cause fraud, then this option clearly is favoured. This is due to its ability to create very large distances between coins, which can be separated by many dimensions.

Chapter 4 - Implementation Systems

In this chapter, the electronics and software developed throughout the course of this research are discussed. The enhanced coin mechanism, which is examined at the start of this chapter, was designed to be reused throughout the course of the research. It was used to provide the capture of data for experimentation, also for an enhancement to the existing Tetrel coin recognition scheme, and finally for implementation of a real time Wavelet coin recognition system.

4.1 DSP Coin Mechanism Development

The basic hardware and software required for a DSP coin recognition mechanism was developed as a modular block. This strategy was chosen as it could be seen that many of the system requirements, for research and experimentation as well as for implementing different recognition schemes, were the same.

The hardware that was produced to form this modular block was based on an existing Tetrel Technology coin mechanism. This basis was chosen as the company has been researching coin mechanisms for over 10 years and had developed a data generation scheme that could produce measurements, which contained both metallic coin composition and dimensional information. This method was modified to produce an improved system that would generate reliable data that was sufficiently characteristic so as to allow differentiation between different coins. It was agreed that this data would form a good basis for DSP research and development. A full description of this system development follows.

4.1.1 Mechanical Development

As was mentioned in the introduction to this thesis, it was desirable that the new coin mechanism would be a size match for the existing coin mechanism. This was achieved, and the basic mechanical overview of how the coin mechanism operates can be seen in Figure 4.1.

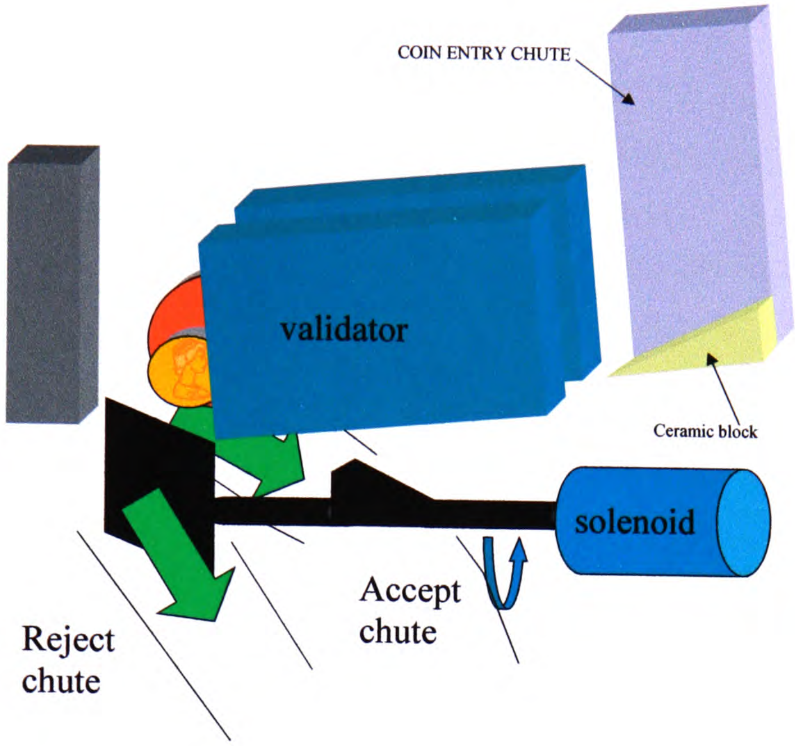


Figure 4.1 – Mechanical overview of the coin mechanism construction

A coin enters down the entry chute and is controlled for bounce by the ceramic block. The coin then rolls down through the validator transducer section due to gravity alone. The transducers are positioned such that the coin will have enough time to settle and roll consistently along the base of the validator without any variation in movement. Exact positioning of the transducers can be seen in Figure 4.2. The inductor positions were repeatedly modified until an optimally different, but consistent waveform for each coin in the target sample coin set was obtained. The selection of the coin set is discussed in Section 5.1.

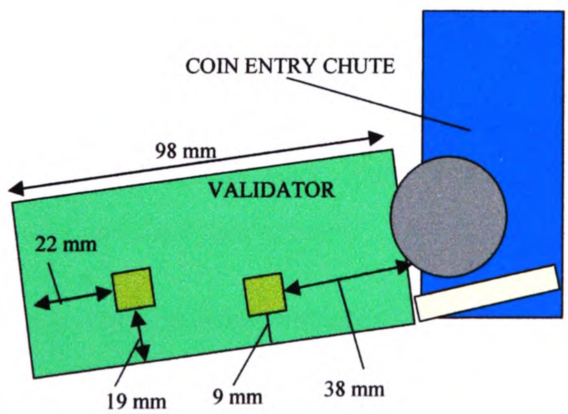


Figure 4.2 – Transducer Positions

4.1.2 Data Capture Enhancement

In this subsection the reader is presented with a discussion of the development of a coin mechanism, this is used as a test bench for DSP algorithms and to obtain test data. A solid place to start such a development proved to be an existing coin mechanism that could be enhanced into a condition where it could be used as a tool for research purposes.

The existing coin mechanisms were embedded into payphones produced by Tetrel. These payphones contain a microprocessor to control the functions of the phone hardware and also used to carry out the data processing and recognition for the coin mechanism. The coin mechanism was not intelligent in any way and simply produced a serial data stream that contained integer count information. The count information varied in accordance with the status of the transducers of the coin mechanism, which in turn were effected by the presence of a coin. The phone processor was already being used to near its full processing capacity and so offered very little opportunity of expansion to implement an alternative recognition solution into the existing scheme.

A simple solution to the lack of spare processing power was to add an extra processor to the coin mechanism making the coin mechanism 'intelligent'. Another solution was to use a full custom designed Application Specific Integrated Circuit (ASIC). This route was not pursued as it was agreed that the time required to design and implement an ASIC solution was prohibitive.

Enhancement proceeded by embedding a small RISC processor into the coin mechanism. The specific processor was selected following a search of the products from various manufacturers. The device, an Atmel AT90S2313, was chosen primarily for low cost but also for low current consumption and a feature rich internal architecture. The device was capable of running at 4Mhz and consuming only 4mA, and with being of a RISC architecture each cycle of the processor clock was one instruction cycle for most processor instructions.

The device also had many other desirable qualities such as its program memory being 'in-system' flash programmable up to 1000 times. This allowed the device to be situated into a coin mechanism and then to have its internal program completely changed, which could be useful both in prototyping and production models. It also possessed 128 bytes EEPROM that was capable of serial re-programming up to 100,000 times. The EEPROM was a perfect area for storing coin limits that may need to be updated several times during the life span of the coin mechanism.

Design for manufacture also featured in the selection of this processor, as the EEPROM could be pre-programmed with coin limits before being inserted into a payphone. This was agreed to be a very useful function as it could, in the future act as an enabler to automated coin mechanism test, coin limit programming and production. This could eliminate wasted time in production programming coin limits or inserting a faulty coin mechanism into a payphone, activities that all presently consume production personnel time.

Another parameter that had to be considered when selecting a processor was the time available for making a recognition decision. The philosophy behind the mechanical handling of a coin in the coin validator was to let gravity move the coin through the mechanism. As the coin exits the mechanism it meets an except/reject control flap, this means that the time available for processing the coin data generated was only 8ms. This time was set from the time the coin finished passing the last transducer in the validator until it met the coin flap. A Reduced Instruction Set (RISC) processor architecture meets the requirements this time limitation places on speed of processing. Other options such as, for example, the 8051 and its derivatives were considered, these could not be used due to the greater number of clock cycles required to perform instructions with these processors. This increase in time could be countered by increasing system clock speed, but then the overall current consumption of the devices would increase, again making them non-viable options.

The inclusion of a processor enabled the coin mechanisms to have a degree of intelligence, performing both data processing and recognition calculations without the assistance of the payphone microprocessor. A block diagram illustrating the basic circuit design can be seen in Figure 4.3 with a full circuit diagram being available in Appendix D.

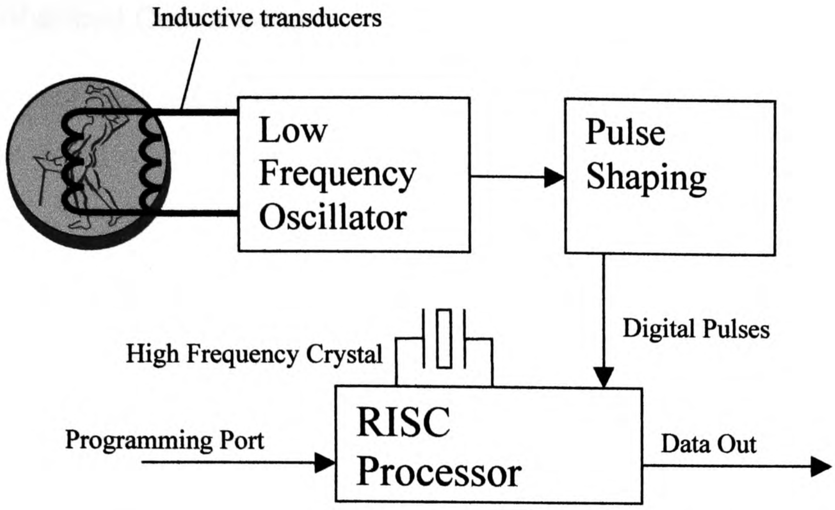


Figure 4.3 – Enhanced Coin mechanism block circuit diagram

In all the existing coin mechanisms, previous to this research, the transducers used were either only capacative, or capacative and inductive. As can be seen from Figure 4.3 only inductive transducers were used in this enhanced version. This was because the capacative transducers were found to be highly susceptible to instability in their readings due to humidity. Humidity would effect the reading of the high frequency capacative coin mechanism, by producing condensation on the coin mechanism PCB. The condensation was found to change the in circuit resistance, producing a variation in the oscillators frequency, and hence producing instability in the coin mechanism readings.

Tetrel had not previously been able to develop a coin mechanism using inductive only transducers. This was due to the high data rates that were necessary to obtain sufficient resolution coin data for recognition from inductive only coin mechanisms. The current payphone microprocessor could not cope with the increased bandwidth of data required. In addition, the recognition algorithms required by the increased number of peaks (up to 4 times as many with bimetallic coins) generated by an inductive only mechanism could not be catered for in the existing payphone processor software.

4.1.3 Enhanced Coin Mechanism Operation

The RISC processor has an internal 16-bit counter that is incremented on each clock pulse from the clock crystal, which is running at 4MHz. The Low frequency oscillator section produces pulses at approximately 8KHz, which are cleaned and shaped to a near digital square wave. The oscillator frequency is modified by two inductive transducers connected in parallel which are sensitive to metals within their proximity. The processor counts 12 pulses from the low frequency oscillator and then stores a count of the crystal clock pulses accumulated in the internal counter. The resultant 16-bit output count is representative of the status of the inductive transducers. The value of 12 pulses was selected arbitrarily, as it delivered a good resolution of coin waveform details. In addition, it kept the data processing throughput requirement from other program functions, such as filtering, to a reasonable level attainable at a system clock speed of 4Mhz.

At switch on the idle frequency of the circuit is taken and stored by the processor which is the current count generated by the circuit after settling has occurred. This count is used to negate the effects of heat and humidity on the low frequency circuit, which has to be inherently unstable for the circuit to be sensitive to the frequency modifications provided by the inductors.

When a coin enters the mechanism, the processor is triggered to start taking readings by the entry opto-transistor. A series of readings is taken until the exit opto-transistor is triggered. The idle count is subtracted from the readings which are taken to give a profile which is representative of the coin which has passed through the mechanism. An example profile is shown in Figure 4.4 (Additional coin waveform samples can be seen in Appendix E).

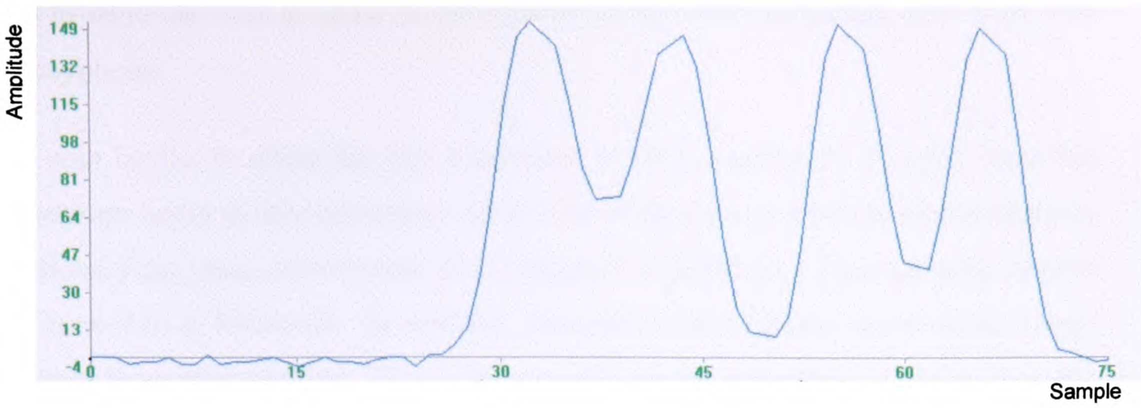


Figure 4.4 – Mexican 1 peso waveform generated from a dual inductive coin mechanism.

The waveform shown above is characteristic of the response of the coin mechanism. An explanation of the shape of the waveform follows, but first it is important to understand the positioning of the inductive transducers. Figure 4.5 shows the transducers positions used to take the coin readings.

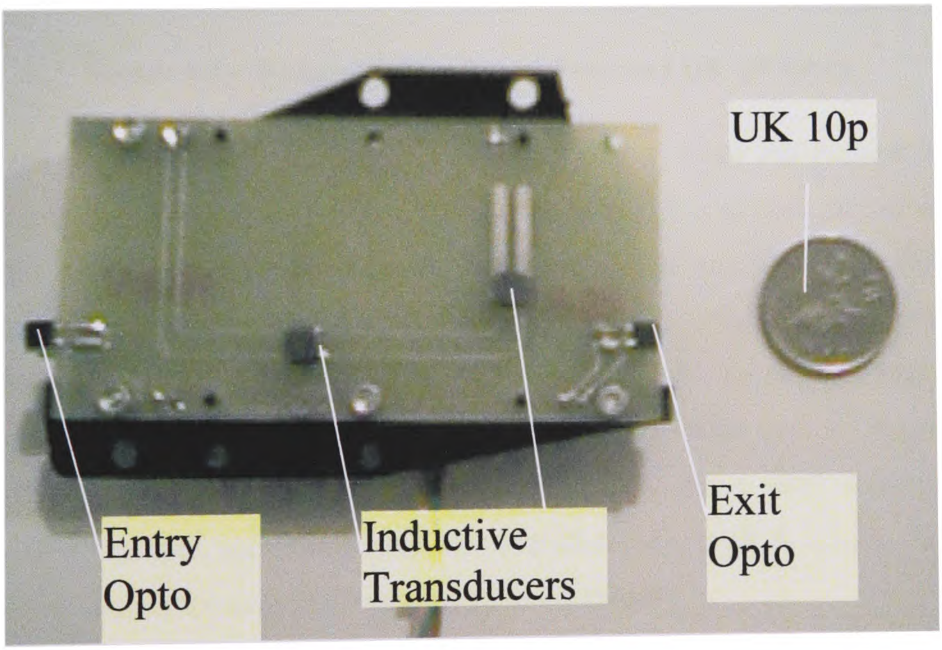


Figure 4.5 – Coin mechanism transducer positions

Observe Figure 4.4, initially the waveform contains no significant data. This is because the coin has triggered the entry opto but has not entered the area containing the inductive transducers. The transducers are positioned at a distance from the entry opto to allow the coin to settle on the runway on the coin mechanism after entry into the payphone.

The coin begins to effect the first transducer at about reading 27 (x-axis). Now the transducers begin to take horizontal 'slices' out of the coin as it passes their respective locations. Four peaks can be seen, this is because the Mexican 1 Peso test coin (shown in Figure 4.6) is bimetallic. In addition, the second set of peaks has a larger trough between them than the first. This is because the second transducer is higher than the first and consequently has more of the centre metal of the coin effecting it.

Figure 4.6 – Bimetallic Mexican 1 Peso and UK 10 Pence

The important features of transducer positioning follow, these are presented in the form of rules that allow positioning of the transducers to give an optimal and different signal for each coin in a given coin set, whilst retaining consistency for the coins. N.B. desirable differences in the coin waveforms can be anything from amplitude, waveform shape, differing waveform peak/trough heights/depths or peak/trough duration's. The reason why any, and all, of these properties can be important is because each of these properties can be characterised by a DSP decomposition method. For example, using Wavelets, the amplitude of a peak in a coin waveform will directly impact the amplitude of a Wavelet coefficient.

Transducer positioning guidelines are as follows: -

- The first transducer encountered along the coin runway must be sufficiently far away from the point of entry to allow the coin to settle on the runway.

- One transducer must be positioned such that it will obtain a slice through any size coin, from the largest to the smallest. This involves placing the transducer close to the coin runway.

- Position the transducer that is close to the runway such that it can optimally detect a change in metal in bimetallic small coins, whilst still detecting the smallest of coins in the set.

- Ensure to utilise the distance between the transducers to provide additional information about the diameter of the coin. This is performed by ensuring that the largest coin in the set will be effecting both transducers together at some point in it's travel past the traducers. Next, ensure that the smallest coin has a period of travel past the traducers where it is having a minimal effect on both transducers. This will ensure that the detection of dimensional difference between coins in optimised.

- Place one transducer higher than the transducer close to the runway. This can be used so that some of the smaller coins do not effect this transducer. This can also be used to increase the diameter detection distance of the previous point.

- Place the highest of the transducers such that a change in bimetallic, large coins is optimally detected.

- Combine any or all of the above points to optimally produce repeatable results for each coin type, whilst retaining the largest waveform difference between each coin type.

In summary, the distinctive characteristics that can be obtained from the transducers are amplitude of peaks and troughs, number of peaks and troughs, and overall waveform shape. The amplitude of the peaks and troughs is effected both by the metallic properties of the coin and also its physical dimensions.

4.1.4 Software

The software for the embedded Atmel processor was written in assembly language as the use of a 'C' compiler was found to be too prohibitive in terms of the actual code overhead size. The Atmel device only possessed 2K of program memory and the 'C' overhead filled over one third of this space before any program code had been added. An additional reason to use assembly language rather than 'C' was the increase in program execution speed that could be achieved by hand programming each processor instruction that would be required. This had the effect that very few cycles of the processor were wasted carrying out instructions that were not needed.

The actual program code was based on a modular design using a type of virtual state machine processing (this is based around a switch in a piece of software which can be used to change the programs functionality based on previous actions). This allowed software reuse into the system, in so far as different recognition methods could be 'bolted' onto the main structure of the program. A state machine type kernel design was used to enable the single tasking processor to be able to carry out multiple functions in an independent and close to parallel fashion. That is to say that an area of code, with a specific function, would carry out a single step of a state machine; This would have a desired action and then would release the program flow to allow other state machines to update their status and carry out alternate functions.

A simplified flow chart indicating the major functions of the program and the implementation of the state machines follows in Figure 4.7.

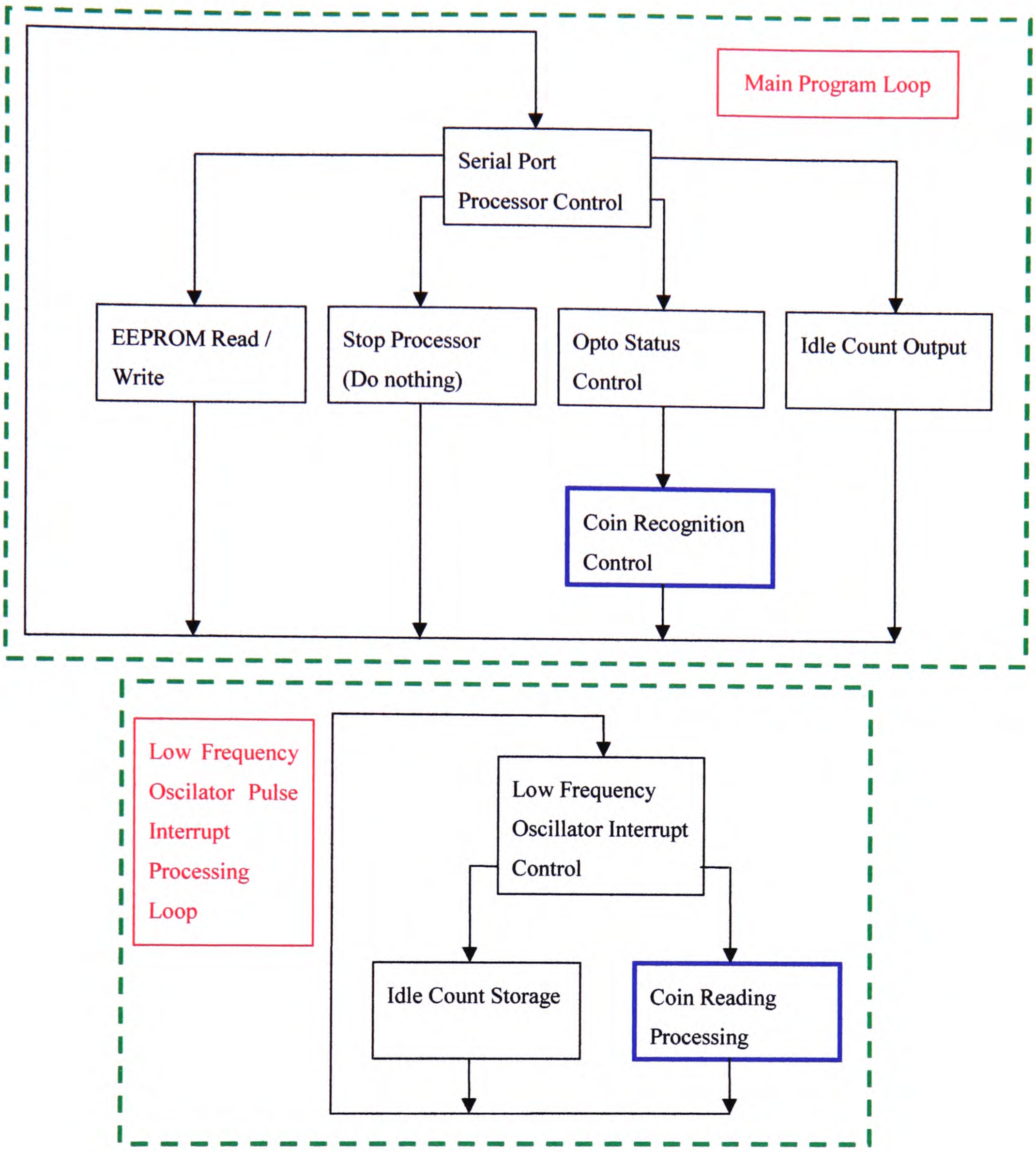


Figure 4.7 – Modular Sections of the DSP coin mechanism main program

The main function blocks that are modified to produce the different versions of the enhanced coin mechanism are shown with blue box. A description of the functionality of each program block follows below.

Serial Port Processor Control – Obtains control codes from the serial port and directs program flow to relevant section.

EEPROM Read /Write – fetches EEPROM data, writes into semi-permanent EEPROM, outputs data to the serial port, and fetches data from the serial port. This is used to store and read coin limit data for valid and invalid coin settings used by the recognition algorithms.

Stop Processor – Stops the Processor

Opto Status Control – Checks the status of the opto transistors and updates their status to internal registers

Coin Recognition Control – Performs any recognition tasks required. This is the area where recognition methods are bolted on to the basic program.

Idle Count Output – Outputs the idle count directly to the serial port. This is used for checking correct operation of the crystal clock and the inductive oscillators.

Low Frequency Oscillator Interrupt Control – This is the entry jump point for the program when an interrupt is generated by the receipt of a low frequency oscillator pulse. It also controls program flow to the following blocks.

Idle Count Storage – stores the idle count. This block is only executed if no opto obstructions have occurred.

Coin Reading Processing – This block can be used to carry out digital filtering and pre-processing of coin readings, such as subtracting the idle count.

4.2 Enhanced Coin Mechanism Monitor - ControlMech

After the enhanced coin mechanism had been developed, there was a requirement to be able to communicate with the device for programming and monitoring the processor functions. The *ControlMech* program was developed to meet this need.

The program was developed to integrate into the authors working operating system environment, which was Windows NT. ControlMech was developed in Visual C++ and provided methods to communicate with the coin mechanism processor through its serial port. The program could also plot graphs of coin waveforms, upload and download EEPROM data, monitor coin mechanism oscillators and optical transistors.

This program was also used to gather coin waveforms for processing during the research.

A screen shot of the program plotting graphs of the coin mechanism coin waveforms is shown in Figure 4.8

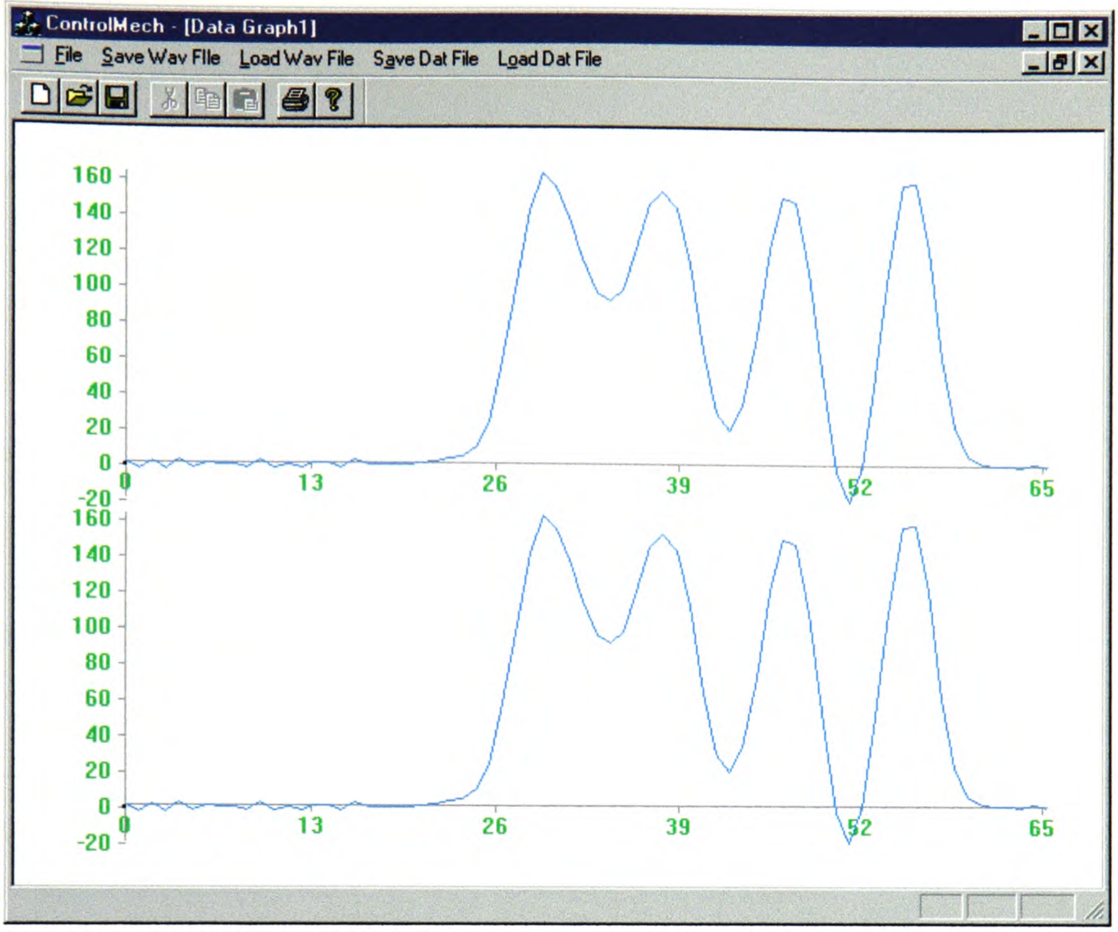


Figure 4.8 – ControlMech screen shot

4.3 TPP1 – The First Steps to DSP Coin Recognition

The embedded RISC processor with additional programming was utilised to provide the first DSP recognition solution, which operated using the amplitudes of peaks and troughs in the coin waveforms. This system was an improvement on the existing Tetrel system, which could only carry out recognition using two readings from either peaks or troughs. The new version could carry out recognition utilising up to eight peak/trough readings from the coin waveforms. This enabled utilisation of more complicated waveforms produced by the inductive data capture mechanism to be utilised by the recognition method. This first method was called TPP1 by Tetrel as this was the designator of the intended product.

TPP1 was an outdoor payphone intended for the Mexican market. This validator was fundamental to the development of a successful payphone for the outdoor environment because it utilised only inductive transducers. This prevented effects from humidity (See section 4.1.2), which had detrimentally effected previous capacative coin mechanisms.

4.3.1 TPP1 Implementation

The basic hardware and software of this implementation is the enhanced DSP coin mechanism as discussed in section 4.1. The modular form of the software in this processor was utilised to deliver the required functionality from this mechanism. The Coin Recognition Control and Coin Reading Processing state machine blocks were both replaced with program code that would carry out the functions required. These blocks are covered in more detail below.

4.3.2 Coin Reading Processing

When a coin enters the mechanism, the entry opto is obscured allowing this section of code to become active when the interrupt is triggered. Two program functions were required from this code, moving average filtering of input counts, and also peak and trough locating within the waveform via turning point detection.

The Moving Average Filter (MAF) function was included as it helped to smooth out the inconstancies in the coin readings. These were caused by minor transient fluctuations in the low frequency oscillator, which had to be inherently unstable for the presence of a coin, next to the inductors, to modify the operating frequency.

The MAF was implemented using a cyclic first in first out (FIFO) buffer. Various different filter lengths were tested for performance in both smoothing and detail loss. Insufficient smoothing and the waveform would have small transient variations throughout its length. Too much smoothing and small characteristic peaks and troughs within the coin waveforms were lost.

A final length of the FIFO buffer, and hence the filter, was chosen to be four integers. This value was chosen as it delivered a good compromise between reading stability and detail loss, also as it enabled fast calculation of the division part of the filter algorithm. This is because division by values of 2^n (such as 2,4,8,16,32) when implemented in assembly language are simple bit right rotation operations. To divide a 16 bit number by two in this way would require two processor cycles (two 8-bit operations) of rotate right. Whereas to do a full divide on an unsigned 16 bit number with the Atmel device would take a minimum of 235 and a maximum of 251 processor cycles Atmel (1997).

A block diagram illustrating the operation of the MAF is shown in Figure 4.9.

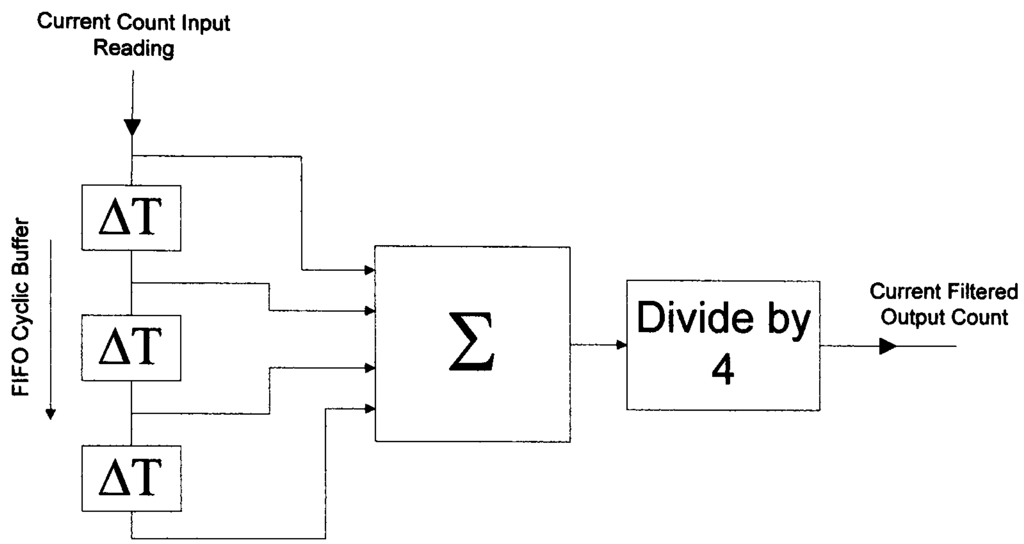


Figure 4.9 – Block diagram of Moving Average Filter

The filter simply adds the current count value together with three previous values stored in the cyclic buffer and divides the result by four to produce a smoothed output value.

The turning point location section operates by monitoring the gradient of the input values to find peaks and troughs in the waveform. A turning point was located when the gradient of the coin waveform changed sign, negative to positive or positive to negative. When the algorithm was tested, it was found that it was locating small fluctuations in the waveform which were not required to be detected as turning points. This was overcome by setting thresholds, both positive and negative, that the gradient had to exceed before a turning point was registered as being valid.

Consider the example section from a coin waveform shown in Figure 4.10, this will be used to illustrate the operation of the turning point location algorithm. Table 4.1 shows a simulation of the algorithm on this waveform using a threshold of two for example purposes only (in the real implementation of this algorithm a value of ten was used which was found by successively testing and adjustment of the value, until a satisfactory result was obtained).

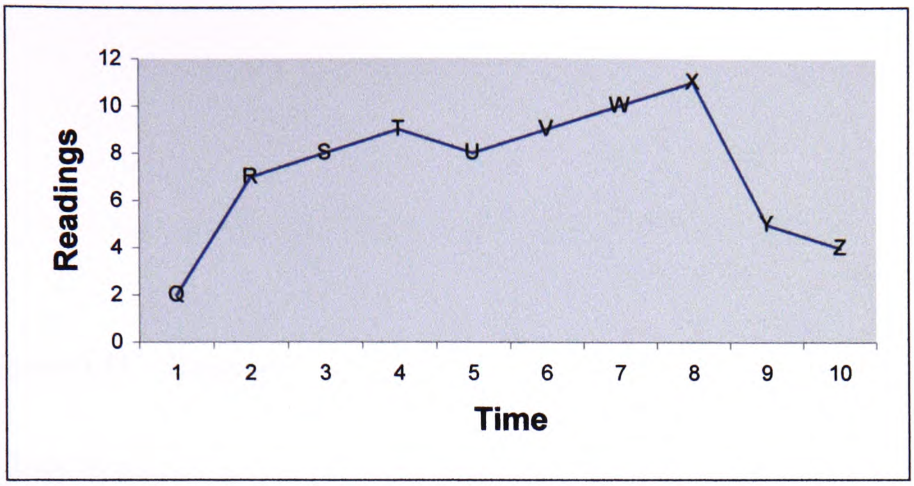


Figure 4.10 – Example coin waveform

Readings	Gradient (X2 – X1)	Positive Threshold Exceeded	Negative Threshold Exceeded	Turning Point Found	Turing Point Valid
QR	5	Yes(QR)	No	No	No
RS	1	Yes(QR)	No	No	No
ST	1	Yes(QR)	No	No	No
TU	-1	Yes(QR)	No	Yes	No
UV	1	Yes(QR)	No	Yes	No
VW	1	Yes(QR)	No	No	No
WX	1	Yes(QR)	No	No	No
XY	-6	Yes(QR)	Yes(XY)	Yes	Yes found at point X
YZ	-1	No	Yes(XY)	No	No

Table 4.1 – Simulation results for turning point algorithm

The table illustrates that the required turning point at X is located correctly and those at T and U, which are not required, are ignored

An actual output from the *ControlMech* program communicating with a processor demonstrating this algorithm is shown in Figure 4.11. The detected turning point locations are denoted by the red blocks.

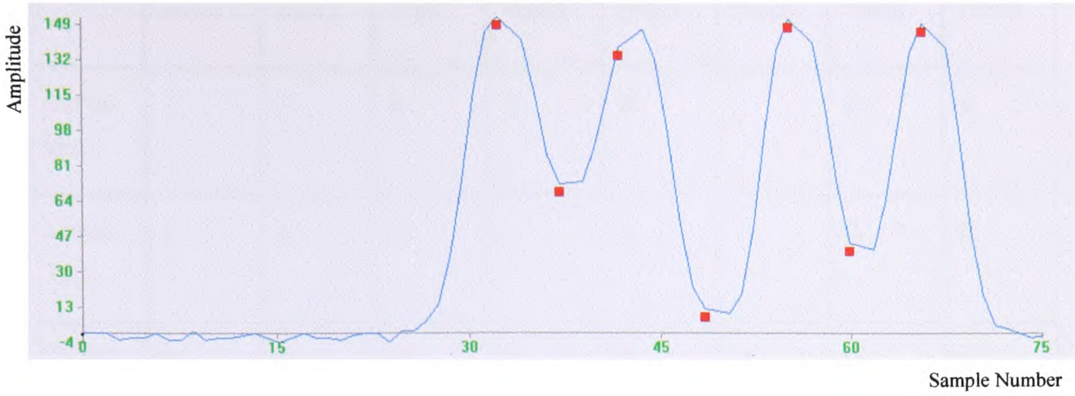


Figure 4.11 – Example of peak and trough detection in coin waveforms

4.3.3 Coin Recognition Control

This code segment was used to check the location of a turning point against the coin limits stored in EEPROM to ascertain if the reading matched any of the limits.

When a turning point was detected by the Coin Reading Processing code the current waveform value was stored for later processing by the Coin Recognition Control state machine.

This state machine was sharing processor time with both the Serial Port Processor Control and the Opto Status Control state machines. By operating in this way, the value of the turning point could begin to be processed and checked against the coin limits before the whole coin waveform had been received. This was to ensure coin recognition would occur before the coin arrived at the accept/reject flap. The functionality of this code segment was split into various steps of the state machine to ensure that other parts of the program would continue to be serviced by processor time.

The current peak being processed (first, second, third, etc. peak within the waveform) would lead to a set of limits being retrieved from the EEPROM these were compared with the current reading. If the reading was within the limits, the result was a binary 1 if not then the result was a binary 0. These results from each compare operation were stored in a bit array, with each row in the array representing a set of limits and each column representing a peak. An example result from this operation can be seen in Table 4.2.

	Peak1	Peak2	Peak3	Peak4	Peak5	Peak6	Peak7	Peak8
Limits Set1	1	1	0	1	0	1	0	0
Limits Set2	1	1	1	1	1	1	0	0
Limits Set3	0	0	0	0	0	0	0	0
Limits Set4	0	1	0	1	1	0	0	0

Table 4.2- Example coin recognition bit array result

The table shows a coin waveform which contained six readings and which was matched the limits of set 2.

The presence of a matching coin was checked for by logically ANDing the rows in the bit array and ignoring coin limit sets that contained more peak limits than were received during recognition.

The recognition cycle is finished when the coin clears the exit opto.

4.4 DSP Investigations

The DSP investigations were conducted, for the most part by developing and testing algorithms using Matlab. The exception to this was the coin waveform test data, which was obtained using a modified version of the enhanced coin mechanism and the *ControlMech* program.

Again, a modular approach was taken to the development of test programs within Matlab. Basic routines for controlling test data and to display results were produced with different data analysis and distance metric techniques being 'bolted-on' to these basic modules.

4.4.1 Data Capture

As mentioned in the introduction to this subsection, the data capture system used a modified version of the enhanced coin mechanism. The modified RISC processor software did not contain a coin recognition state machine module. The Coin Reading Processing module was modified to output a data stream containing waveform readings, with the idle count subtracted, onto the serial port of the processor. The coin mechanism was mounted into an existing payphone chassis, as shown in Figure 4.12, which ensured a reliable coin delivery action.

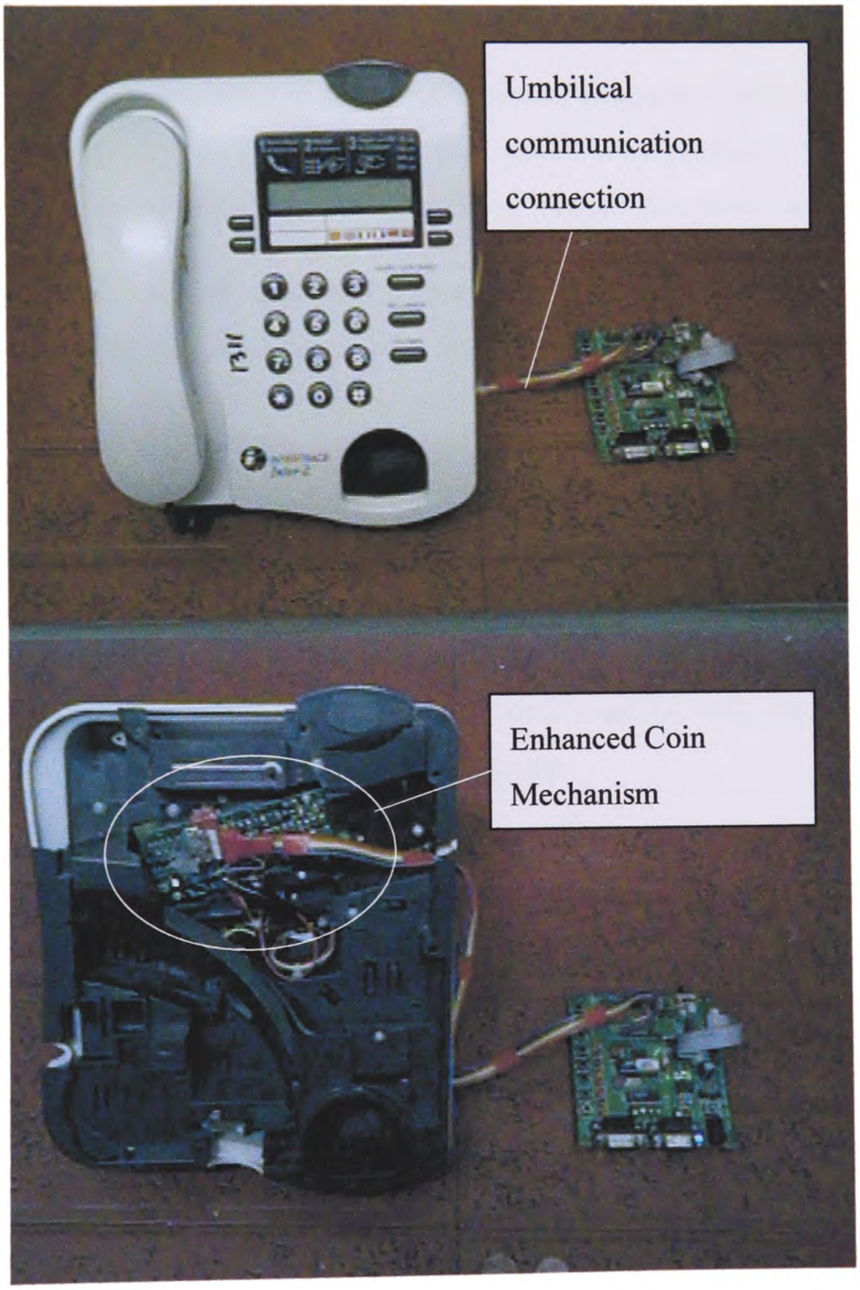


Figure 4.12 – Enhanced coin mechanism mounted into a payphone chassis

The *ControlMech* program had a section that could receive the coin waveform data. On receipt the program plotted a graph of the data for inspection and saved a file to disk in comma delimited ASCII integer format. *ControlMech* prompted for a string that was to represent the type of coin at the start of a data saving session. This was the name under which each coin file was saved with the addition of a postfix number, which was incremented, after each coin sampling was taken. Using this system it was a simple task to very quickly accumulate a large set of sample waveforms for each coin that was used in the DSP research.

4.5 Data Processing and Mining Software Development

A modular approach was taken to the development of programs within Matlab. Basic routines for controlling test data and to display results were produced, with different data analysis and distance metric techniques being 'bolted' onto these basic modules.

The Matlab scripts are presented and discussed in the order in which the research proceeded. However, to understand how these Matlab scripts/programs would be applied to the data, a flowchart is presented in Figure 4.13. The flowchart contains the order in which the scripts would be used to conduct the analysis, at each stage annotations of scripts that can be used are shown. The reader is directed to observe this flowchart at this point, and to reference back to it during the discussion of the individual scripts/programs.

Note: The Matlab scripts are available on the disk supplied with this thesis or by Email from the author at Sharman@technologist.com.

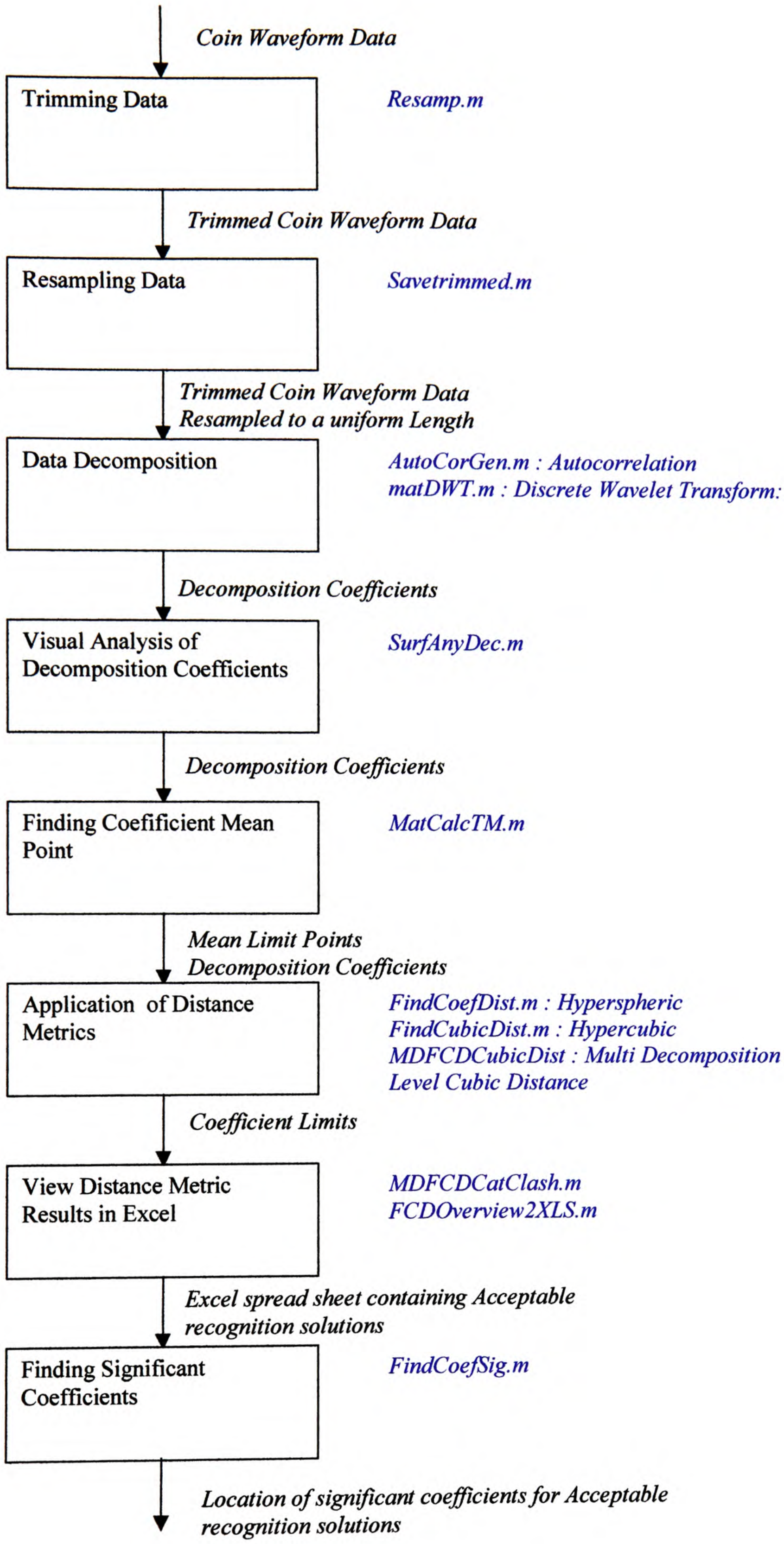


Figure 4.13 - Flowchart showing application order of Matlab Files to Data

4.5.1 Autocorrelation Coefficient Generation

The program *AutoCorGen* was written to fulfil this function. As mentioned in the background theory Section 3.1, autocorrelation is a simple process where a waveform is multiplied against shifted versions of itself and the resulting coefficients are summed to give each autocorrelation number.

The algorithm was passed over all the target coin waveform data files for all possible shift positions. The resultant data was stored in large arrays, one for each coin.

4.5.2 Production of Discrete Wavelet Transform Coefficients

In order to use the data mining technique (see Chapter 5) it is necessary to have a bulk load of transform coefficient data with which to test hypothesis. *MatDWT*, a Matlab program, was designed that would process the coin data files generated from the *ControlMech* program and generate DWT coefficients.

The Wavelets selected based on the discussions raised in the Background theory chapter (see Section 3.6 Page 24). Ranges from four different families encompassing 37 Wavelets were chosen. These particular Wavelets were used as they offered a good cross range of different Wavelet shapes available in Matlab.

The Wavelets were selected by following one of three criteria. Either the Wavelet was desired to be similar to the coin waveform shapes with low frequency, rounded, curves, or with a high frequency, sharp profile. The third criterion was included to help with the final implementation of the Wavelets in a recognition system and dictated that the DWT coefficients should be computed with a low computational effort.

To exhaustively analyse the coin data files with each Wavelet type, and to ensure all the data was extracted and processed with each Wavelet, a target of ten decomposition levels was used. This value was chosen as a trade of between finding the lowest data processing requirement, whilst ensuring that even with the smallest of the Wavelets, the coin waveforms were decimated to a point were the Wavelet was of a comparable, or larger, size to the original coin waveform. This had the effect that very little or no information was being gained after the tenth level of decomposition that was of any worth in determining the coin type.

The result from the *matDWT* program acting on the coin waveform data was 74 data files, which contained DWT coefficients for the approximation and detail results for each of the 37 filters. Each data file possessed a Matlab structure array that contained 3600 coefficient arrays. One array for each coin's DWT results of the 20 coin data samples transformed to 10 levels of decomposition for each of the 18 coin types. This gave an overall number of 266400 coefficient arrays. Which gives a clear indication of what a large processing task this was.

To understand this further and as an example of a Matlab script observe the following script code. The code which is prefixed by a % is a comment in Matlab language and has been used to explain the program operation in detail. This piece of code is repeatedly run for each filter in the target set of 37.

```
%calculates DWT for a filter set by curfilt variable
%to a depth of 10 decompositions

clear; %clear the enviroment
Curfilelist;%load a list of the coin data types

%write name of current filter into out array
out = struct('filter',Filters(curfilt).Name);

%iterate through coin types
for file = 1:numtypes
   %iterate through coin samples
   for curdat = 1:Files(file).numTest  %this loop steps through the coin types
      %construct target file name
      targfile = [ datdir Files(file).name int2str(curdat) '.dat'];
      clear Temp;
      Temp = load(targfile); %load a coin sample
      datnum = int2str(curdat);

      %Calculate Transforms
      for declevel = 1:10
        %calculate the Wavelet decomposition
         [TempA,TempD] = dwt(Temp,Filters(curfilt).Name);
         dec = ['d' int2str(declevel)];

         %store approximation coefficients
         s = [ Filters(curfilt).SName '_' Files(file).name '_dint2str(declevel)
'_a' int2str(curdat)];
         out = setfield(out,s,TempA);

         %store detail coefficents
         s = [ Filters(curfilt).SName '_' Files(file).name '_d' int2str(declevel)
'_d'  int2str(curdat)];
         out = setfield(out,s,TempD);

        %prepare for next loop
         clear Temp;
         Temp = TempA; % make approimation the root of the next Wavelet Transform
         clear TempA;
         clear TempD;
      end % declevel

   end % curdat

end % file

savename = [ datdir Filters(curfilt).SName filelistnumber '.mat' ];
%save the DWT coeficients
save(savename,'out');
```

4.5.3 Visual Analysis of DWT Coefficients

A three-dimensional plot of the DWT coefficients was achieved using *SurfAnyDec* a Matlab program. This program plots 20, three dimensional graphs for a selected Wavelet type, each containing the results from the DWT coefficients generated from 10 waveforms of one coin type, produced at all 10 decomposition levels. This gives a very clear picture of the action of the DWT and how variations in similar coin waveforms effect the transform. An example of the results obtained from this program, along with a discussion can be found in Section 6.2 Page 78.

4.5.4 Finding Coefficient Mean Limit Point

Regardless whether the distance metric used Hyper-spheres, Hyper-elliptics or Hyper-cubes, one common factor between all methods was that the mean point of the analysis coefficients was required. The Matlab program *MatcalcTM* was produced to fulfil this function.

Initially a trimmed mean was used to provide robustness to outliers in the data. This method was discontinued and replaced with a standard mean during the latter stages of the research. This was because during the testing of the final algorithm using a Hyper-cubic approach all coin samples were required to be used as templates for setting limits and the outliers were found to be important in providing limits which would encompass all coins.

4.5.5 Implementation of Distance Metrics

The distance metrics (discussed in Section 3.7 page 28) were initially implemented to test coefficients from the approximation and detail results of the DWT separately and one dimension at a time. Matlab programs were produced to test the hyper-spheric and hyper-cubic metrics, and these were *FindCoefDist* and *FindCubicDist* respectively.

The programs both operated in very similar ways, and processed all the coefficients generated from the 37 test Wavelets used to calculate the DWT test data. The results were produced in the form of large arrays which contained a list with each row containing the filter type, decomposition level, approximation or detail coefficients, the number of clashes of either the hyper-cubes or hyper-spheres, and finally the minimum distance which was observed between the hyper-cubes or hyper-spheres.

To view the distance metric test results a Matlab program *FCDoverview2XLS* was written that used Dynamic Data Exchange to communicate with Excel. Using this program, the results were transmitted from the Matlab array to an Excel spreadsheet where they could be viewed and sorted in order of number of clashes and minimum distance. This allowed observations about the effectiveness of the distance metrics to be easily made.

4.5.6 Combining DWT decomposition level results

During testing both the hyper-spheric and hyper-cubic distance metrics were found to be failing to find a set of dimensions which would be able to completely separate the set of test coins. To counter this it was necessary to combine results from multiple decomposition levels. This had the effect that the number of dimensions, which were used to represent a coin, was increased which offered a greater possibility of locating dimensions in which the coin set would not clash. For example if a coin waveform was decomposed into: -

Decomposition 1 = (1,2,3)

Decomposition 2 = (4,5)

The co-ordinates of the coin reading, when decompositions 1 and 2 were combined would become 1,2,3,4,5. Once combination of the decomposition level coefficients was achieved, the distance metric method was implemented in the same way as when only one decomposition level was involved.

The simplest way to test for a possible solution would be to combine all ten available decomposition levels together and test if a solution existed. This was not done, as it would not yield the combination of levels that would deliver the solution that required the least amount of processing power.

To test for the simplest solution the decomposition levels where combined in order of ever increasing computational effort. For example, firstly only two decomposition levels were combined. This was done starting with the quickest DWT coefficients to calculate i.e. 1&2, then 2&3 continuing on up to 9&10. Then combinations of 3 decomposition levels were tested starting with 1&2&3, then 2&3&4 and so on up to 8&9&10. This procedure was continued until all 10 decomposition levels were combined together, with a total of 55 combinations being possible.

A complete map of the combinations of decomposition levels can be seen in Figure 4.14. The red squares indicate which decomposition levels were used in each combination set.

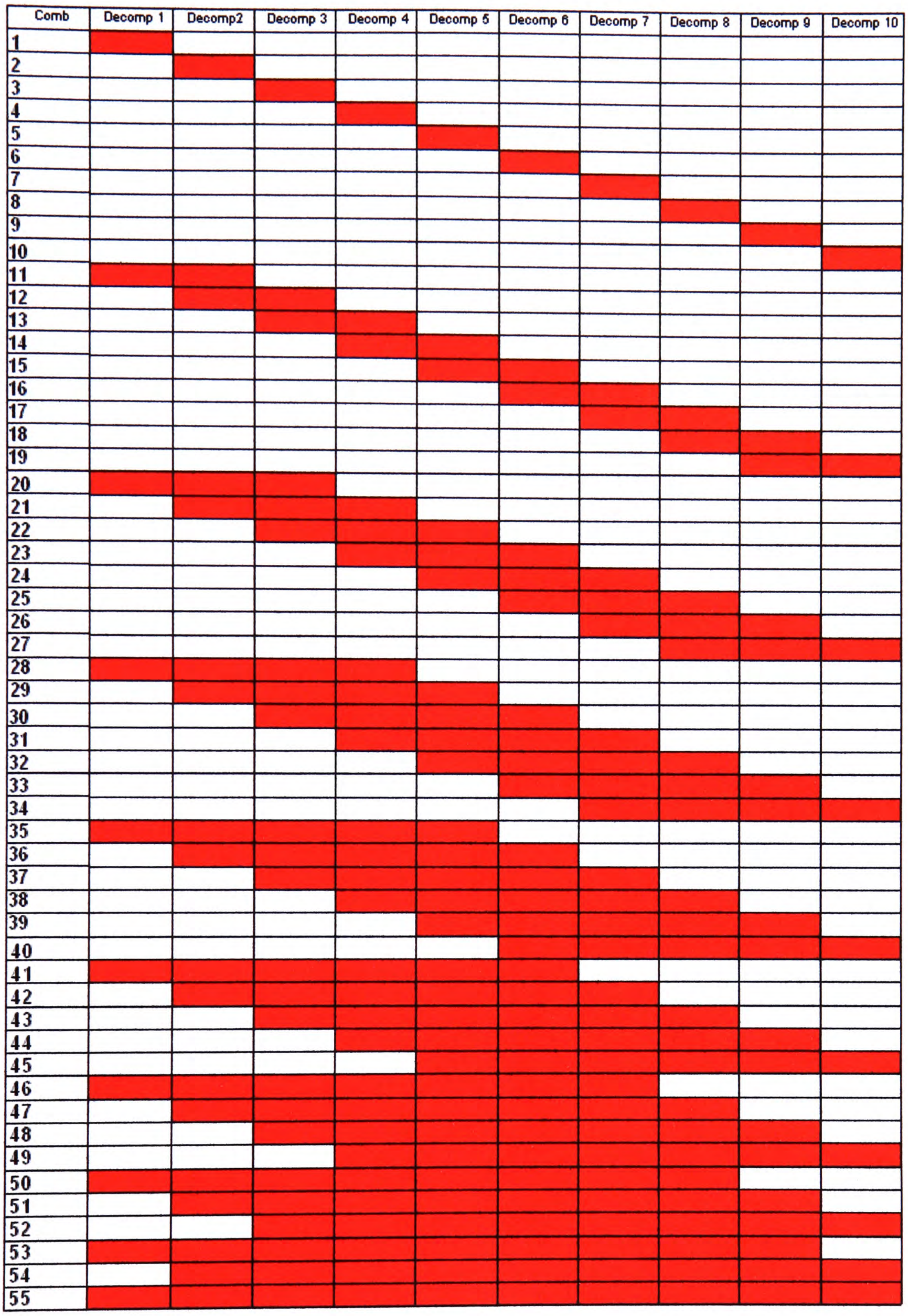


Figure 4.14 – DWT decomposition level coefficient combinations

4.5.7 Trimming and re-sampling of data files

In this sub-section the novel method of pre-processing input coin waveforms is developed. This is the way in which the coin waveforms were pre-processed before being processed by the Wavelet transform, for the purpose of temporally normalising the data. This normalised property is required by the Wavelet transform and not, for instance by the peak and trough spotting algorithm, because with the DWT temporal information is also encoded and becomes a feature of the output coefficients.

Observe the coin waveforms stored by the data capture mechanism for the same coin (ten examples of the same coin can be seen Figure 4.15), this shows that the waveforms are slightly dissimilar, in amplitude and either shifted in time, dilated, contracted or a combination of all three. This is a feature of the speed of the coins travelling through the data capture mechanism, which was discussed in section 4.1.

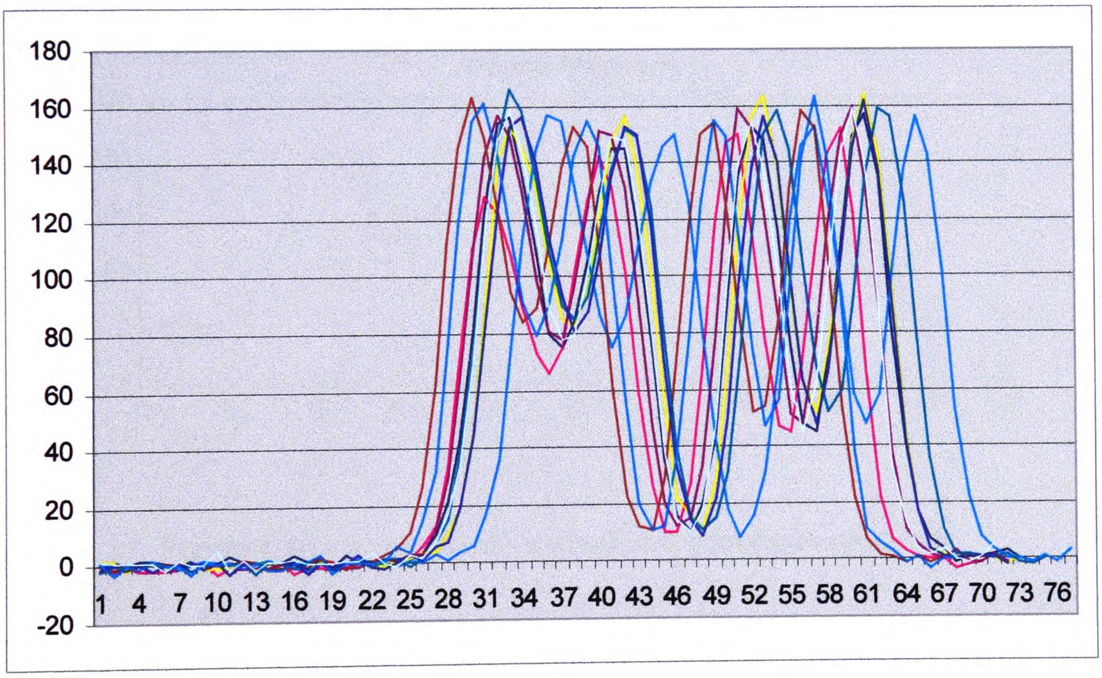


Figure 4.15 – 10 captured coin waveforms from Mexican 10 Peso coin type

The amplitude differences were expected and were allowed for by the coin limits set around the DWT coefficient results. It was initially hoped that the time effects would have disappeared from the DWT coefficients as successive decimations, during each level of decomposition, filtered any differences together.

The filtering effects did draw the waveforms together, but not enough for consistent results to be observed in the DWT coefficients. A new approach was required which would pre-process and clean up the waveforms before the DWT processing began.

The pre-processing took the form of trimming and re-sampling the waveforms to ensure a constant length. The trimming action removed any extraneous data that existed before and after the waveform, see Figure 4.16.

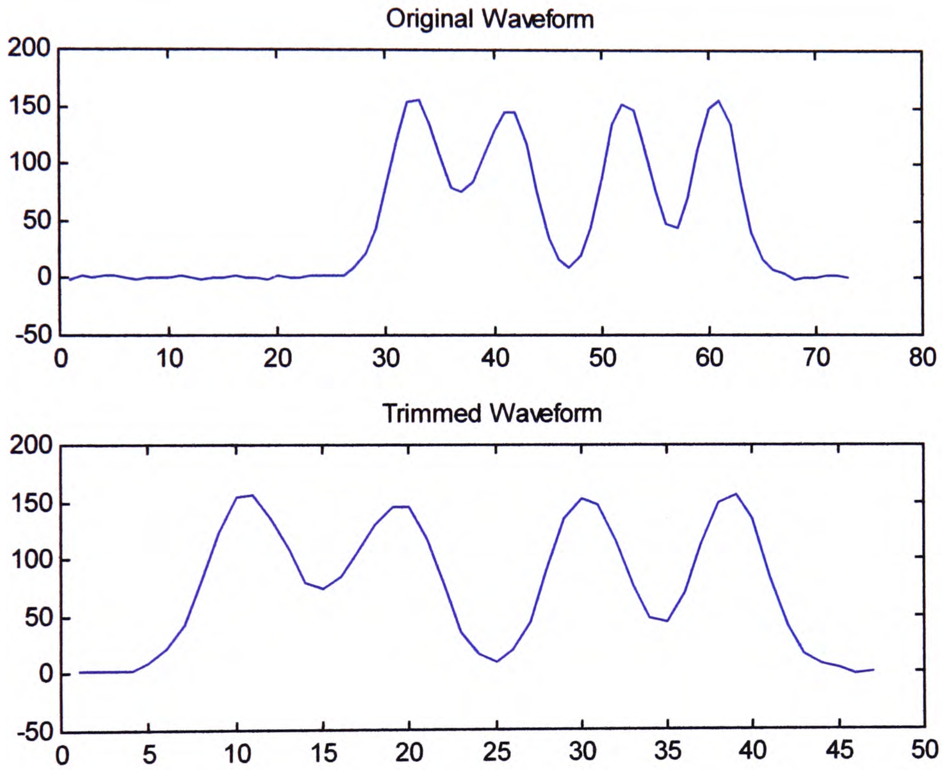


Figure 4.16 – Example of original and trimmed waveform

After the extraneous data had been removed, the peaks of the waveforms still were not aligned. This is where the re-sampling of the data comes to the fore. When re-sampling the data, a waveform size was chosen that was bigger than any of the coin waveforms in their trimmed form. This method was used so that no data was lost from the waveforms during the process. The value that satisfied this requirement for the test sample data was 60 samples. To reach this value the waveforms were interpolated and filtered, then decimated and filtered.

The effect this has on the data originally considered at the start of this sub section can be seen in Figure 4.17.

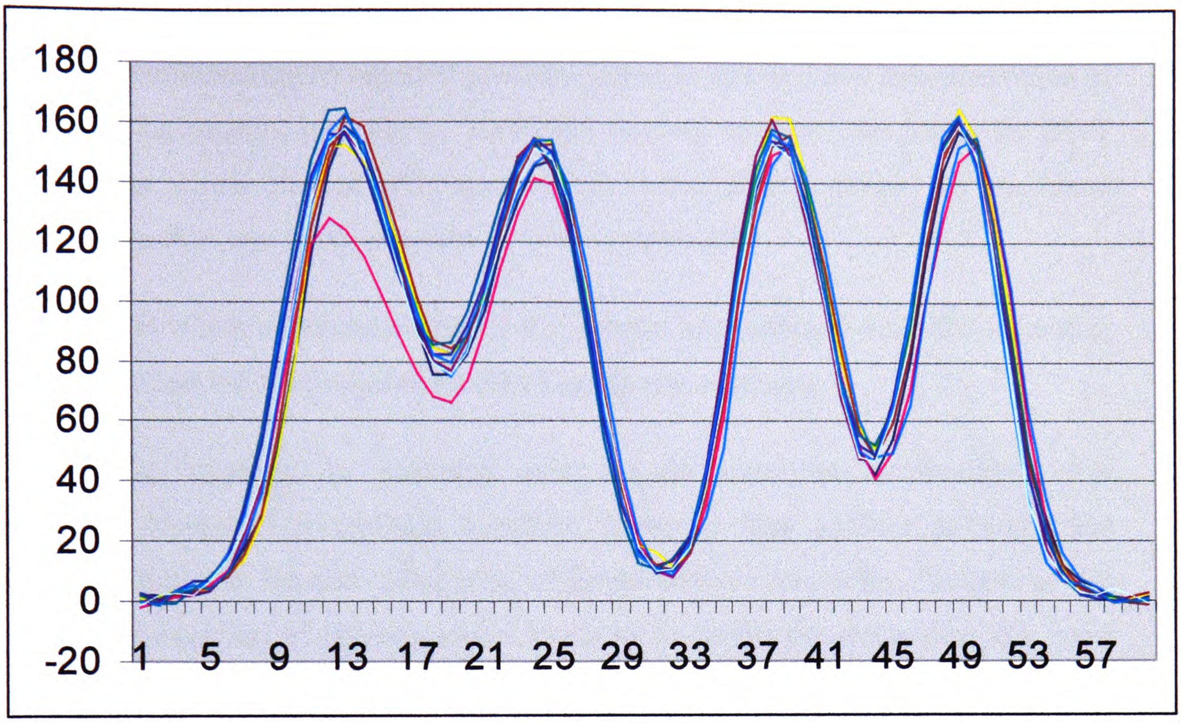


Figure 4.17 – Trimmed and resampled coin waveforms from Mexican 10 Peso coin type

Dynamic Time Warping (DTW) was also considered to carry out this function as the author had previously had experience of this during research involving speech recognition. DTW could be applied in this case by warping certain section of the waveforms until its peaks were located within a regular intervals. Due to the heavy computational requirement of DWT the simpler method, involving a lower computational requirement and which was working successfully, was used. The interested reader can consult Owens (1994) for more information on DTW.

4.5.8 Finding Significant Recognition Coefficients

After a successful decomposition method and distance metric combination had been found, only one piece of information was missing which prevented the development of an efficient real time implementation of a coin recognition system. That information was which DWT coefficients were significant, from the decomposition level combinations used, for recognition to be possible.

This information was useful in finding what the bare minimum computational effort was, while still making recognition possible. Consider that in a real implementation of a recognition system, it would be inefficient to store and use coin limits for DWT coefficients which clashed with many or all coins, or that offered no additional information than was being provided by other coefficients.

To discover which coefficients offered the highest information content for separating the target coins the Matlab program *FindCoefSig* was produced.

FindCoefSig operated by scanning over the decomposition combinations that produced successful recognition solutions. It would then produce an array that contained one row for each comparing of limits between two coins. Remember that for the combination of decomposition levels to be useful for recognition this must produce at least one coefficient dimension in which there was no limit clash.

Then *FindCoefSig* found the location of each coefficient that had limits that did not clash between the two coins. For every occurrence of this it would store a '1' in the array. This was then repeated for each combination of two coins, which for 19 coins in the Target sample set would mean 171 pairs (see Table 3.1).

When this operation was completed each column in the array was summed which gave a value which indicated the 'usefulness' of each coefficient in comparing the coins i.e. how many times each coefficient had managed to separate the coins.

Next, a new array was constructed column by column from the original array in descending numerical order. As each column was added the rows were summed across and the result was checked to see if each row was above zero. This was done to see if enough coefficient columns had been added to cover all the pairs of coins.

The final result was a minimum list of significant coefficients required for each decomposition combination that delivered a recognition solution.

4.6 Implementation of Real Time Wavelet Coin Recognition System

To gain a complete understanding of the benefits and problems of developing a real time coin recognition system this final investigation tool was developed. The philosophy behind the development of this system was to, as closely as possible, link the developed algorithms to a real time coin mechanism implementation.

The way in which this was achieved was to modify a version of the *ControlMech* software to emulate the action of a processor performing the Wavelet recognition calculations. This software was called *WaveActiveX*, the ActiveX portion of the name will become apparent in the following text.

As the algorithms for the Wavelet recognition system existed in Matlab, the natural way for the program development to proceed was to embed Matlab into the *WaveActiveX* program. The programming method that could deliver this embedding requirement was using ActiveX.

ActiveX, developed by Microsoft, is "A set of technologies that enables software components to interact with one another in a networked environment, regardless of the language in which they were created." (Microsoft)

A block diagram of the whole system is shown in Figure 4.18, with a simplified flowchart of the software operation within the system shown in Figure 4.19.

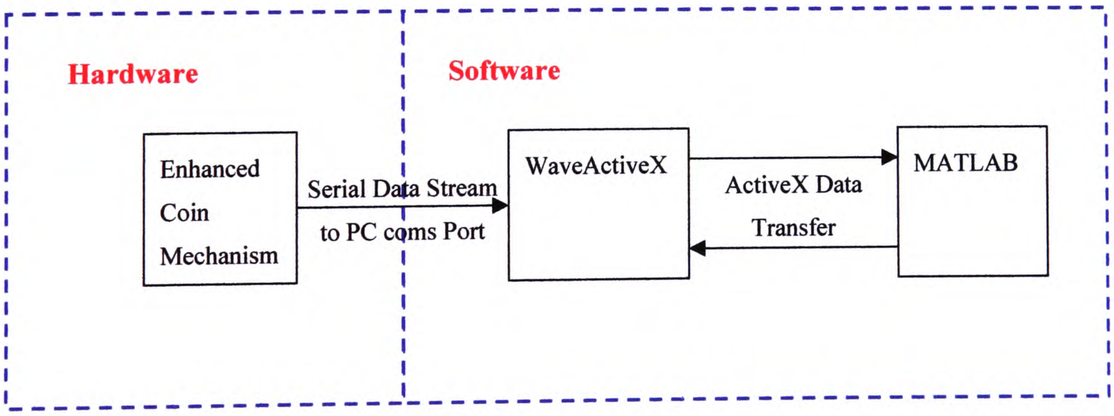


Figure 4.18 – *WaveActiveX* system configuration

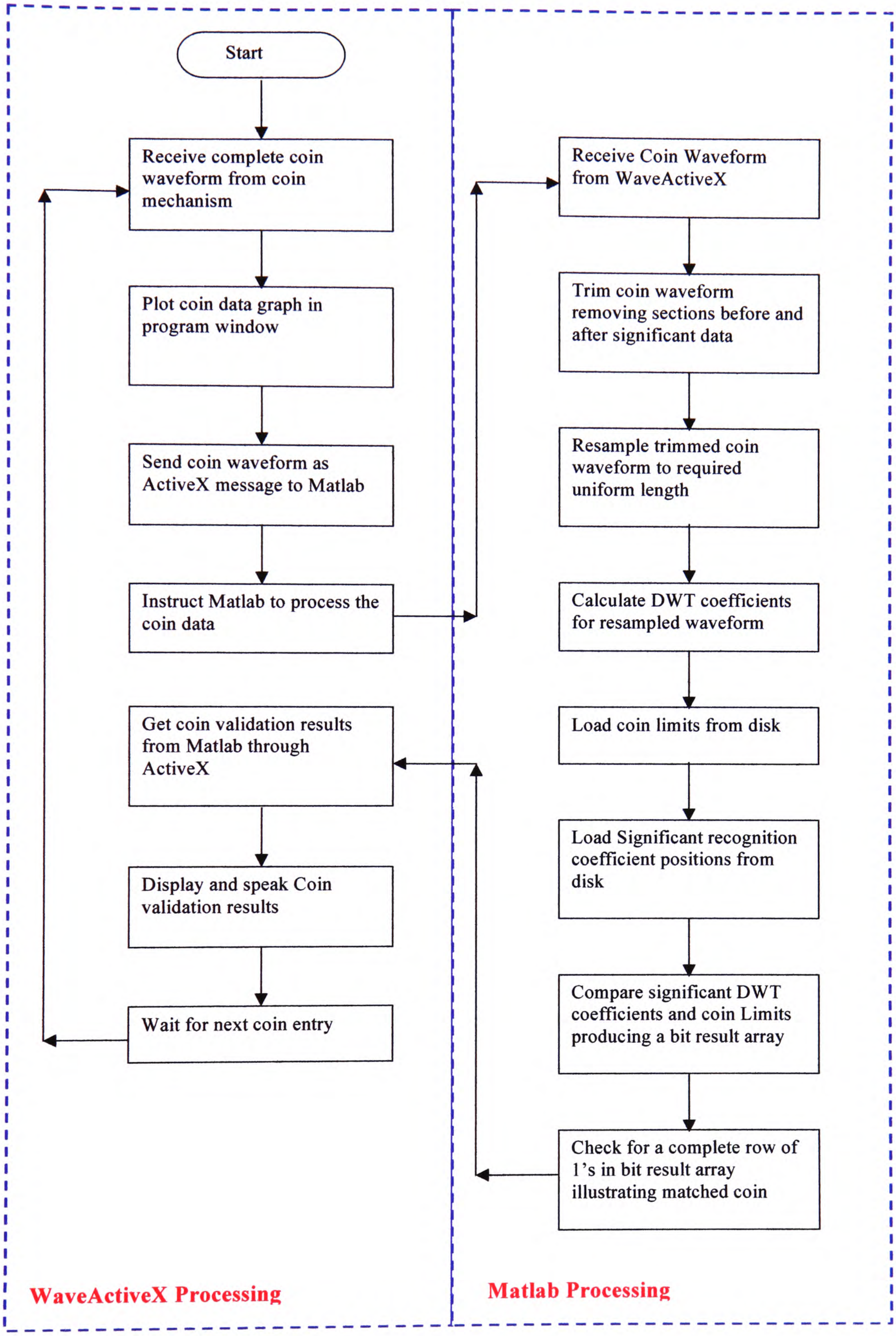


Figure 4.19 – WaveActiveX software operation flowchart

Chapter 5 - Data Processing and Analysis

This chapter contains details of the methods followed for carrying out analysis using the systems developed in the previous chapter. Firstly, the philosophy of the kinds of data that were captured is studied. Following this, there are two types of analysis examined in detail. The main analysis of interest in this thesis is the Wavelet results, but the Autocorrelation method is also included as this is used as an example set of results in section Chapter 6 .

5.1 Generation of Test Data

The data capture system, described in the systems chapter, was used to collect 20 samples of 18 coin types. These coin types were selected to produce a good cross range of the possible outputs from the coin mechanism, so that the DSP algorithms would be robust, and operate for many coin types. The collection included Steel Copper and bimetallic coin types, these would produce positive and negative deflections from the idle frequency of the coin mechanism as well as combinations of 3 to 7 peaks and troughs in the coin waveforms. Another coin selection criterion was not only to produce a wide range of differing waveforms, but also waveforms which were similar in shape and amplitude from different coins. This would allow assessment of the ability of the developed algorithms to separate possible fraud coins and coins which would normally cross correlate with a desired acceptance coin in an existing coin mechanism. This data was used throughout the analysis.

The target coins and basic details of the data set are as shown in Table 5.1.

Coin	Number of Peaks	Number of Troughs	General Characteristics
Mexican 1 Peso	4	3	Positive deflection
Mexican 2 Peso	4	3	Negative deflection
Mexican 10 New Peso	1	2	Negative deflection
Mexican 10 Old Peso	1	2	Negative deflection
Mexican 20 Peso	1	2	Negative deflection
Mexican 5 Peso	4	3	Positive deflection
Mexican 50 Centavo	1	2	Negative deflection Large flat centre peak
US Quarter Dollar	1	2	Negative deflection
UK 1 Copper Pence	1	2	Negative deflection Troughs have the same amplitude
UK 1 Pound	1	2	Negative deflection Flat on centre peak
UK 1 Steel Pence	2	1	Positive deflection Two long flat peaks
UK 10 Pence	1	2	Negative deflection Waveform can be noisy
UK 2 Copper Pence	1	2	Negative deflection
UK 2 Pound	1	2	Negative deflection
UK 20 Pence	1	2	Negative deflection Low amplitude noisy waveform
UK 50 New Pence	1	2	Negative deflection
UK 50 Old Pence	1	2	Negative deflection
UK 2 Steel Pence	1	2	Positive deflection Two long flat peaks

Table 5.1 - Target coin data set and coin features

5.2 *Autocorrelation analysis*

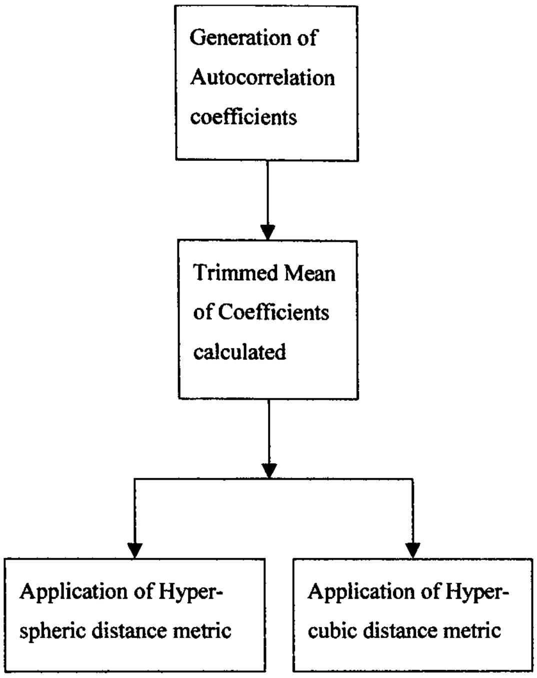


Figure 5.1 – Analysis plan for Autocorrelation investigations

The Autocorrelation investigations proceeded by utilising the Matlab modules, that were discussed in the systems section, as shown in Figure 5.1.

The captured coin waveforms for each coin were processed into autocorrelation coefficients (see section 4.5.1), the trimmed mean of the autocorrelation coefficients was then found, as discussed in section 4.5.4, for each coin by combining the coefficients at each level of correlation. For example, considering the UK 2 copper pence, all the R(1) coefficients were combined for all 20 samples of this coin and the mean found, then the same procedure was repeated for R(2) and so on until all coefficients for this coin had been processed. The process was then repeated for all the other 18 coin types.

After the trimmed mean had been found the distance metrics from section 4.5.5 were applied, the results of this analysis are shown in section 6.1.

5.3 *Wavelet Transform Analysis*

5.3.1 Using Original Data

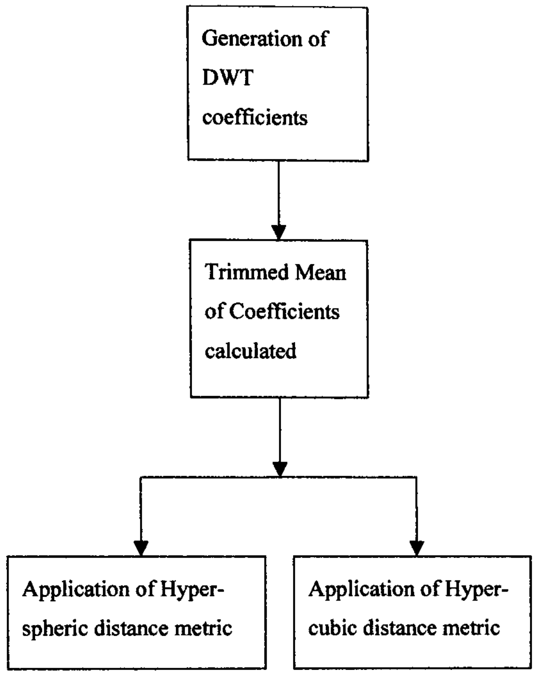


Figure 5.2 – Analysis plan for Wavelet Transform Coefficients – Original Data

The analysis of the Wavelet transform coefficients initially followed a very similar route to that of the autocorrelation values as can be observed from the plan shown in Figure 5.2.

Using the method discussed in section 4.5.2, the DWT coefficients were generated for each of the coin waveforms using 37 different Wavelets to a level of 10 decompositions. The waveforms used were the original data set produced by the data capture system, completely unmodified in any way. The approximation and detail results from both sides of the Wavelet tree were stored for analysis.

A trimmed mean coefficient set was calculated, using the software discussed in section 4.5.4, for each coin from the coefficients stored in each of the 266,400 decomposition result arrays. The DWT coefficients together with the trimmed mean data were then tested with the distance metrics (see section 4.5.5).

The corresponding results are given in Section 6.3.

5.3.2 Using Resampled Data

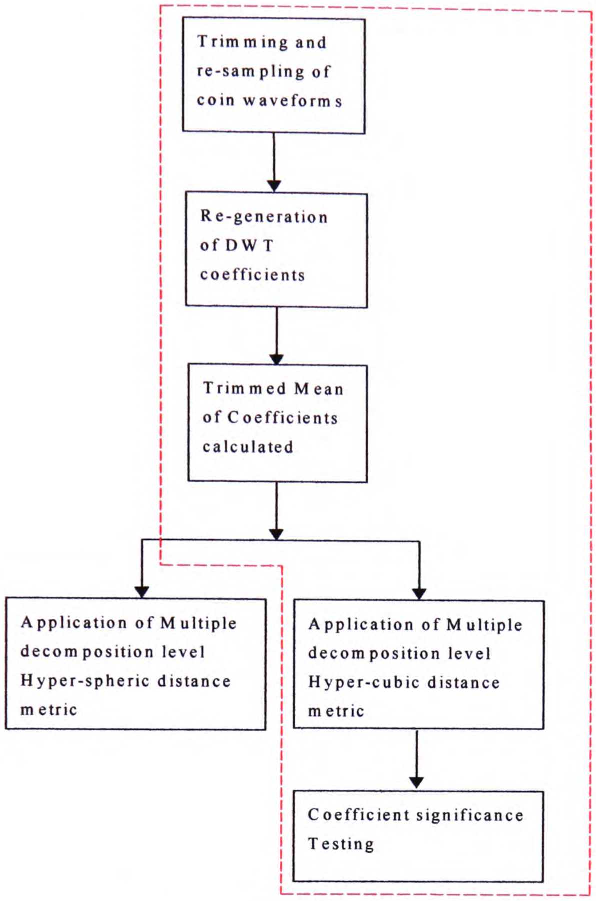


Figure 5.3 – Analysis plan for Wavelet Transform Coefficients – Resampled Data

The analysis route for the resampled data was similar to that followed for the data that had not been pre-processed. The exception with this analysis was that the DWT coefficients had been completely regenerated with the resampled data. In addition, the distance metric analysis tools which were developed had been improved, and supported the Multiple decomposition level approach. A graphical view of the combinations of decomposition levels is given in Figure 4.14 on page 63.

Once a successful result had been obtained with the Hyper-cubic distance metric coefficient significance testing was undertaken.

The results for this analysis are shown in section 6.4.

The area in Figure 5.3 encompassed with the dotted line, is so indicated because this is where viable final recognition solutions were available from.

Chapter 6 - Discussion of Results

In this section analysis is made of Autocorrelation and Wavelet transform results. The autocorrelation results are included so that a full appreciation of the data processing problem can be perceived, and why the more complex method of Wavelets had to be employed.

6.1 Autocorrelation

The autocorrelation tests resulted in a complete failure to separate the coins.

Neither the hyper-cubic or hyper-spheric distance metrics could produce any significant distances between the data, either with or without re-sampling the waveforms. The reasons for this can be seen by visually examining the autocorrelation results. Figure 6.1 illustrates the similarity that was found in the autocorrelation functions for coin waveforms of different coin types, with each coin type shown with a different colour on the graph.

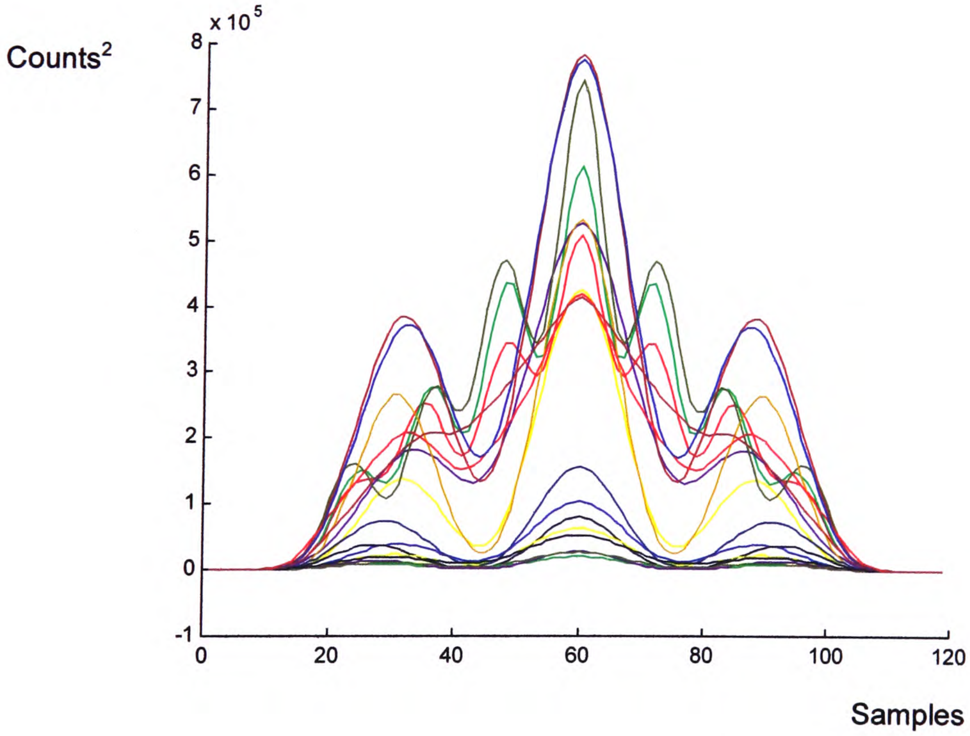


Figure 6.1 – Autocorrelation of each coin type

The coin mechanism, however, does not produce consistent waveforms for repeated passes of a coin. Therefore, when considering the possibility of using autocorrelation for producing distances between coins it is important to regard many samples of the coin waveforms. This is shown in Figure 6.2 again, with each coin type shown with a different colour on the graph.

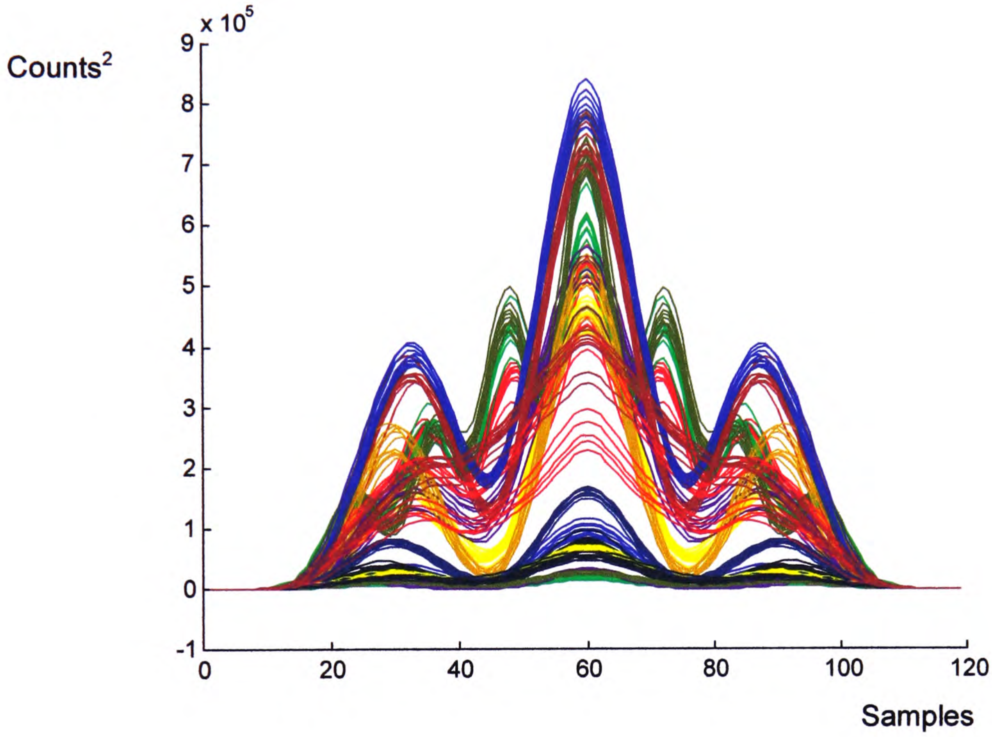


Figure 6.2 – Autocorrelation of 10 samples of each coin type

Now it can be seen how completely inappropriate autocorrelation is for analysing the waveforms. Using only visual inspection, it can be seen that throughout the waveforms there are no areas available where limits could be placed to separate the coins without clashing with other coin waveforms.

Passing the coins through the hyper-cubic distance metric confirms this fact, the results are shown in Table 6.1.

	M_1_Pe	M_2_Pe	M_10_Ne_Pe	M_10_Ol_Pe	M_20_Pe	M_5_Pe	M_50_Ce	U_Qu_Do	U_1_Co_Pe	U_1_Po	U_1_St_Pe	U_10_Pe	U_2_Co_Pe	U_2_Po	U_20_Pe	U_50_Ne_Pe	U_50_Ol_Pe
M_1_Pe																	
M_2_Pe	-5265079																
M_10_Ne_Pe	-1897162	-1995018															
M_10_Ol_Pe	-4044768	-4128798	-1944965														
M_20_Pe	-4706272	-4780243	-2025774	-4261715													
M_5_Pe	-5794263	-6023382	-2533001	-4689978	-5359070												
M_50_Ce	-1777835	-1875212	-720916	-1822135	-1902510	-2412502											
U_Qu_Do	-5491658	-5561136	-2484137	-4731719	-5391571	-6233911	-2364469										
U_1_Co_Pe	-4333268	-4394712	-2005895	-4193084	-4515262	-4959012	-1889673	-5098475									
U_1_Po	-2366872	-2457917	-1057503	-2423790	-2504270	-2994302	-943153	-2971858	-2500433								
U_1_St_Pe	-4215091	-4276127	-1474871	-3569007	-4118022	-4823932	-1358377	-4706300	-3853990	-1949008							
U_10_Pe	-1423482	-1523860	-364373	-1459596	-1541360	-2061157	-305465	-2000933	-1523724	-577417	-1001964						
U_2_Co_Pe	-5737271	-5827905	-2591189	-4836152	-5528772	-6566804	-2471050	-6737696	-5198289	-3077402	-4878451	-2108177					
U_2_Po	-1783822	-1882182	-724603	-1822408	-1906133	-2419099	-642356	-2364807	-1885420	-938279	-1362354	-299437	-2472581				
U_20_Pe	-1517782	-1617455	-459946	-1555100	-1637284	-2154970	-402187	-2096431	-1620520	-674182	-1098078	-188331	-2203827	-394927			
U_50_Ne_Pe	-1470419	-1570588	-411982	-1507024	-1589169	-2107906	-352321	-2048178	-1570692	-624457	-1048951	-162216	-2155568	-347134	-233349		
U_50_Ol_Pe	-1655426	-1754637	-596021	-1691783	-1775720	-2291608	-530021	-2233860	-1754700	-808035	-1233927	-233472	-2341770	-533421	-328230	-281238	
U_2_St_Pe	-4204943	-4269011	-1248191	-3354337	-3960177	-4801146	-1129214	-4607039	-3610714	-1714407	-3545258	-776703	-4805517	-1136329	-870697	-823563	-1008349

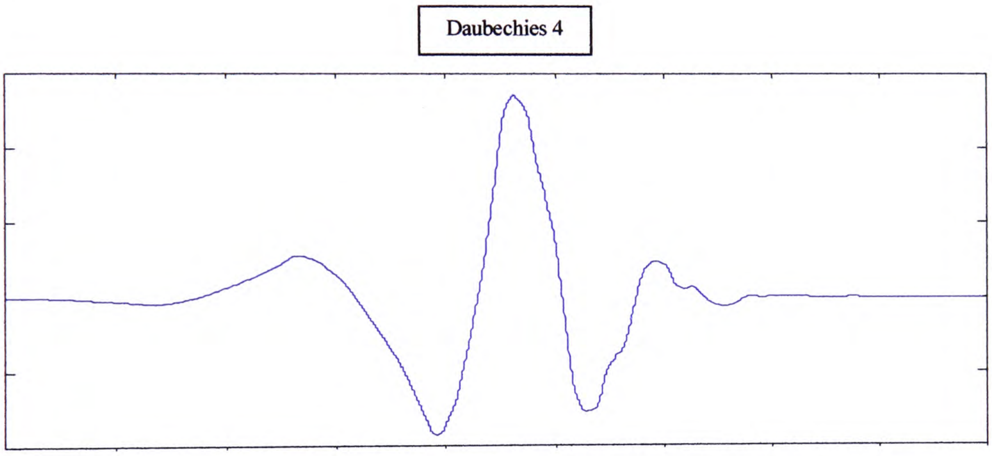
Table 6.1

The coin types are shown in the left-hand column and across the top of the table, the cell contents is the minimum distance that was found between hyper-spheres of each coin type. The Table Entries shown in red are the hyper-spheric clashes that were found. All the crossing points between the coin types are shown in red which means that no coins where able to have a completely separate area defined in the n-dimensional function space used by the distance metrics.

6.2 *Visual Analysis of DWT Coefficients*

These results have been included to give the reader an insight into the action of the Wavelet transform.

Visual analysis of the Wavelet coefficients was conducted during the research using the *SurfAnyDec* Matlab Program. Two example Wavelets will be used to show results from this program, the Daubechies 4 and Coiflets 1 Wavelets, as shown Figure 6.3. These Wavelets were chosen for this demonstration because of their very different shapes, and hence the very different results that they produce. A discussion of the properties of these Wavelets is contained in Section 3.6 on page 24.



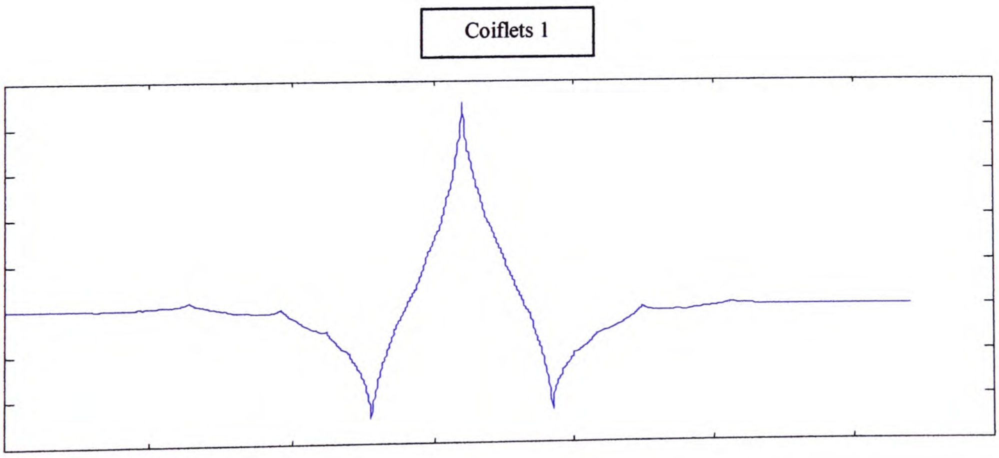


Figure 6.3 – The Daubechies 4 and Coiflets 1 Wavelets

During the analysis the same Mexican 2 Peso coin waveform in its resampled form will be used for all DWT coefficient results to allow the action of the DWT to be clearly observed. A plot of the Mexican 2 Peso coin can be seen in Figure 6.4. Ten samples of the waveform are used so that the action of the Wavelets on slightly different samples can be observed.

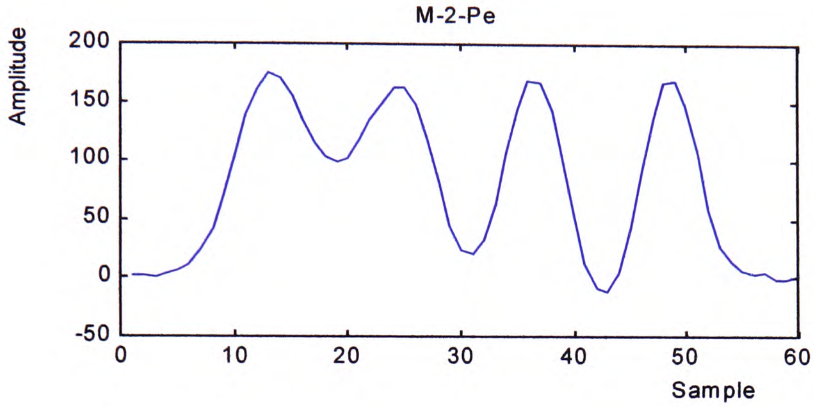


Figure 6.4 – Mexican 2 Peso coin waveform

The results for the decompositions using the Wavelets will be presented first, with a discussion of the results following each set of diagrams. In the interests of brevity only decomposition levels 1,3,5,7,9 are shown. The diagrams show the coefficient number along the x-axis, amplitude of the coefficients on the y-axis and the 1 to 10 coin samples along the z-axis as shown in the example Figure 6.5.

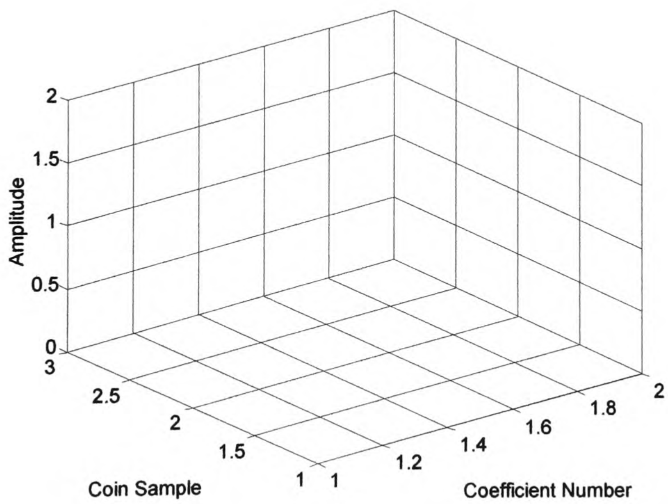
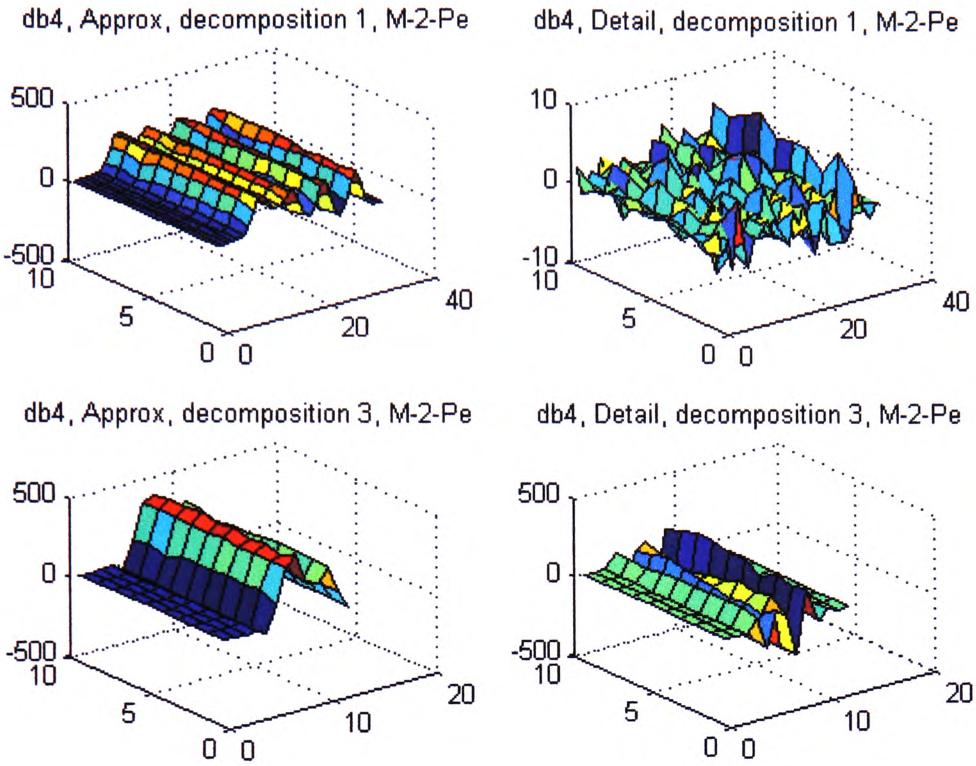
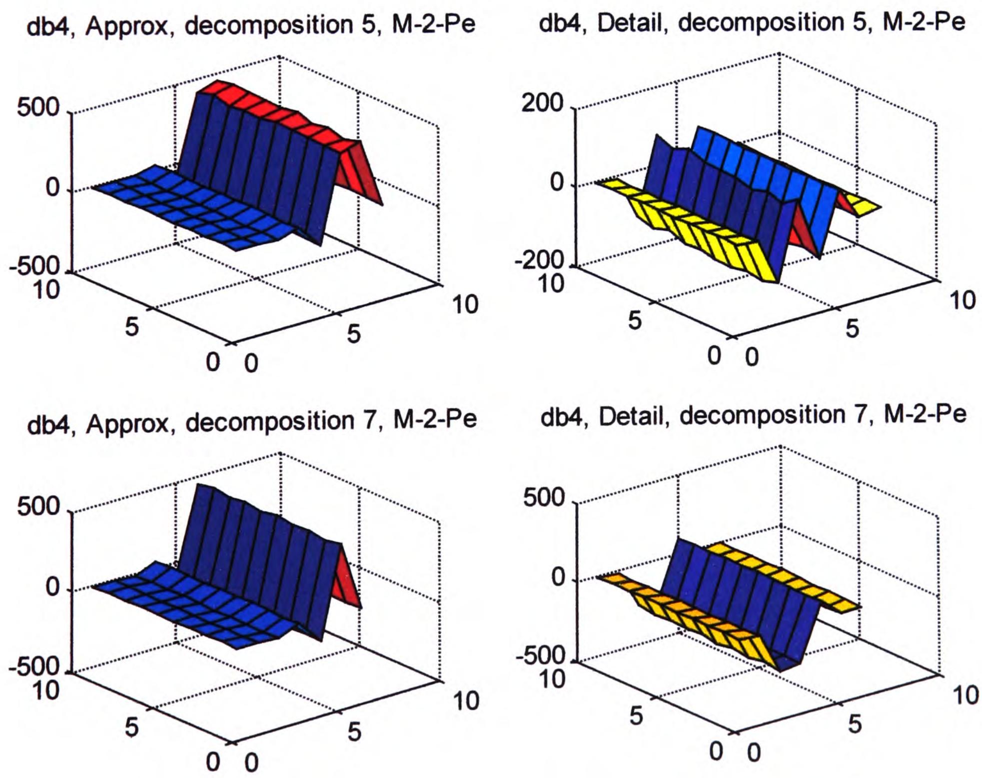
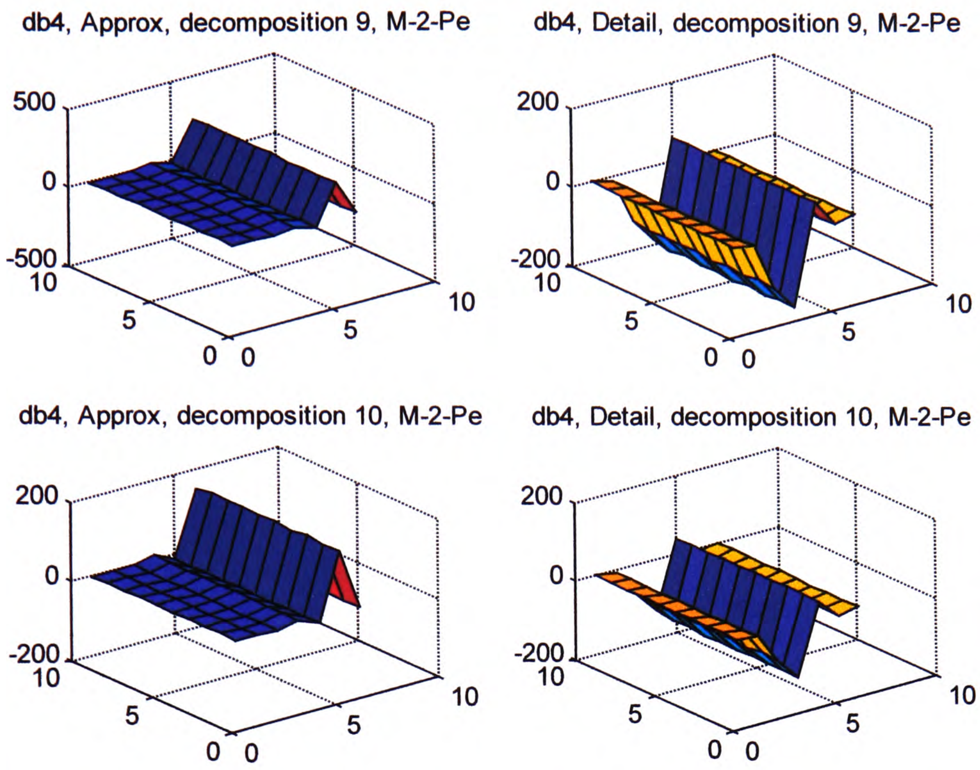


Figure 6.5 - Example Plot Showing Axis Labels

First Considering Daubechies 4 (abbreviated to db4 in the diagrams).

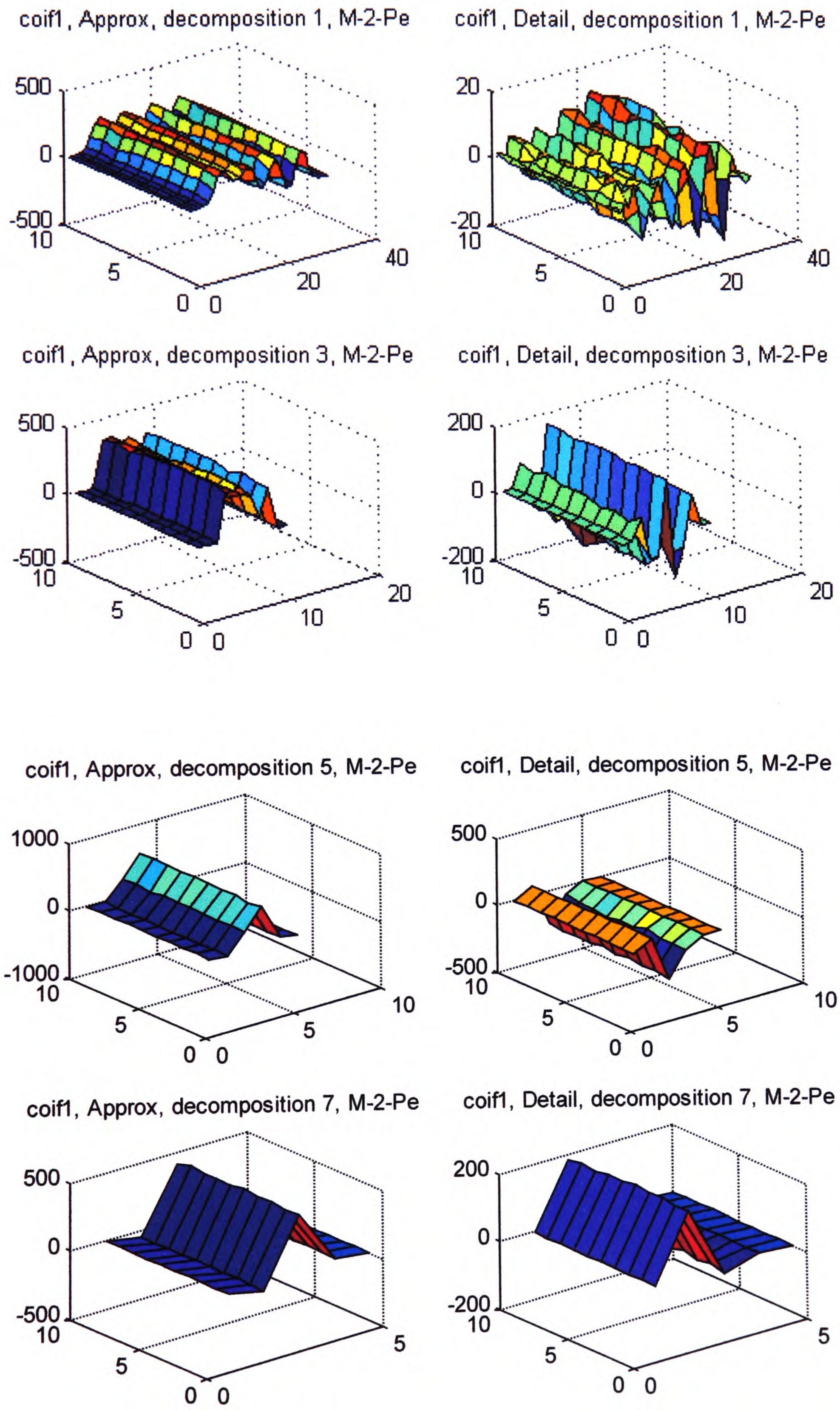


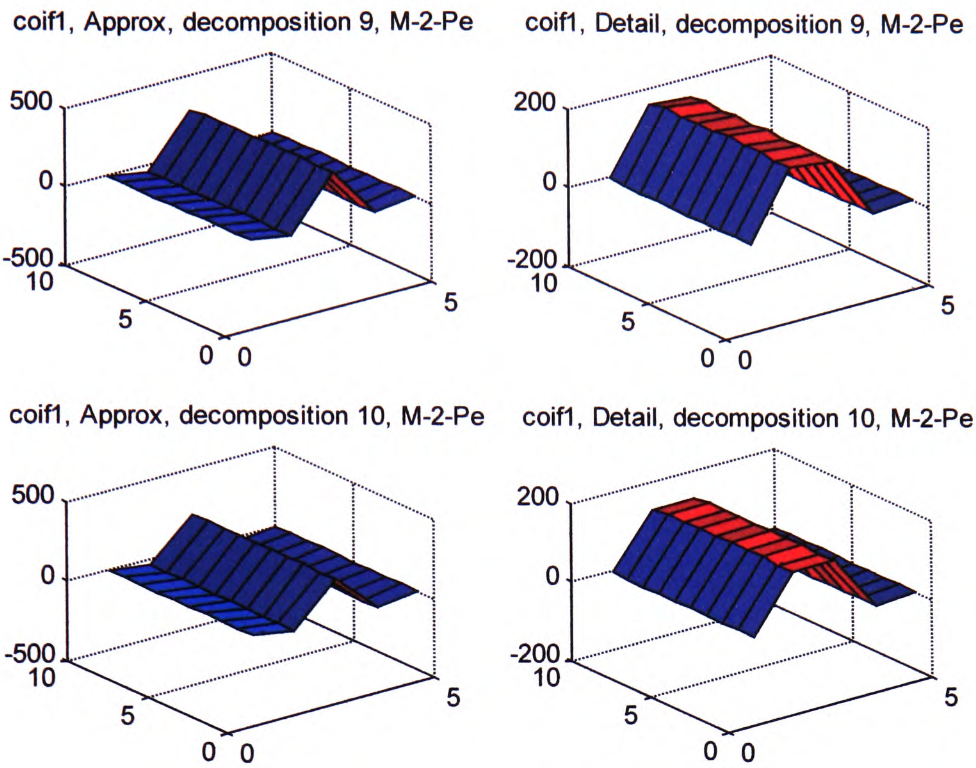




Observing decomposition 1 it can be seen that the Wavelet has managed to encode very little of the data from the waveform with the majority staying the approximation coefficients. By the third decomposition the Wavelet has become sufficiently large, when compared to the data, to begin encoding. The coefficients from the tenth level of decomposition show that the Wavelet has 'drained' the coin waveform of most the high frequency data that it contained, and now only the very slow changing attributes of the signal remain.

Now considering the Coiflets 1 (abbreviated to coif1 in the diagrams).





The action of the Coiflets 1 Wavelet behaviour can be seen to be completely different from the Daubechies 4 as early as decomposition 1. In decomposition 1 the Wavelet has already begun to encode significant amount of the waveform data into detail coefficients, and had resulted in a substantially smoothed approximation waveform. At the fifth decomposition most of the high frequency information of the signal has been removed with only the low frequency data remaining. To achieve the same result the Daubechies 4 needed all ten decomposition levels. Interestingly, the resultant waveforms at decomposition 9 from both Wavelets bare no resemblance to each other, which is a clear indication of how differently the Wavelets have decomposed and encoded the waveform data.

This means that each Wavelet will present a completely different encoding of the data in its coefficients. This has the effect that each set of coefficients from different Wavelets will present a new and different view of the data when used for recognition.

6.3 *Wavelet Transform – Standard Data*

6.3.1 Hyper-spheres and Hyper-cubes

The original coin waveforms were very inconsistent in where the peaks and troughs appeared in the waveforms. This inconsistency was carried though in to the Wavelet coefficients, so profound was the effect that none of the combinations of the 38 filters and the two distance metrics that were implemented could produce a distance between all the coins. This meant that a recognition solution could not be generated using this system configuration.

An example of the type of results obtained is illustrated in Table 6.2 (see Appendix E for coin name abbreviations used in the table)

The table shows the distances obtained between coin hyper-spheres from the approximation coefficients after four decompositions using a Haar Wavelet.

The red cells show the clashing hyper-spheres of which there are 90 out of a possible 171 coin pairs.

This result is typical of the problems that were found in using coin waveforms that had not been pre-processed.

	M_1_Pe	M_2_Pe	M_10_Ne_Pe	M_10_Ol_Pe	M_20_Pe	M_5_Pe	M_50_Ce	U_Qu_Do	U_1_Co_Pe	U_1_Po	U_1_St_Pe	U_10_Pe	U_2_Co_Pe	U_2_Pe	U_2_Po	U_20_Pe	U_5_Pe	U_50_Ne_Pe	U_50_Ol_Pe
M_1_Pe																			
M_2_Pe	-409.417																		
M_10_Ne_Pe	354.5929	236.1002																	
M_10_Ol_Pe	340.3831	218.2475	-321.622																
M_20_Pe	420.3714	292.9505	-233.684	-660.357															
M_5_Pe	-128.275	-335.914	449.2928	421.0073	487.139														
M_50_Ce	373.8956	256.0898	-239.163	-250.443	-163.928	472.0191													
U_Qu_Do	814.6674	690.4801	154.8866	-226.709	-291.229	894.3076	210.8832												
U_1_Co_Pe	445.7596	325.6958	-220.712	-570.774	-541.018	535.518	-161.603	-229.16											
U_1_Po	441.7658	328.1779	-208.013	-330.548	-239.238	548.1009	-156.223	119.5946	-257.955										
U_1_St_Pe	-225.807	-355.422	198.3954	189.6072	266.6019	-116.365	211.1123	633.0916	277.8625	282.4122									
U_10_Pe	370.6194	250.292	-109.314	-123.385	-42.9559	466.4507	-102.283	320.9977	-42.2971	-28.9345	195.6074								
U_2_Co_Pe	485.6924	365.4262	-183.128	-593.084	-662.559	572.0611	-121.121	-485.999	-568.285	-222.8	320.0435	-0.22575							
U_2_Pe	737.8593	620.3335	70.19249	-334.619	-391.631	830.651	131.7932	-212.758	-312.135	25.57531	574.2598	254.266	-659.443						
U_2_Po	388.2863	266.3768	-228.159	-261.785	-181.743	477.153	-197.703	193.9631	-173.603	-144.499	221.1049	-95.9031	-132.757	123.1362					
U_20_Pe	371.9331	252.5604	-90.883	-105.291	-24.5498	470.6709	-85.3307	334.0807	-28.0679	-15.9217	194.2813	-102.947	14.07269	267.9979	-76.9487				
U_5_Pe	358.9067	241.4117	-23.4901	-32.0131	50.59918	462.9	-16.2132	412.1566	48.83656	53.84968	182.1167	-30.4809	90.82444	343.823	-2.48093	-33.314			
U_50_Ne_Pe	363.9319	243.833	-157.952	-170.503	-88.7055	458.8283	-149.572	281.4952	-84.7841	-72.7018	193.7393	-122.561	-43.0984	211.3278	-141.49	-102.18	-32.2744		
U_50_Ol_Pe	376.0003	258.0908	-219.709	-225.759	-138.854	473.2645	-199.712	242.9746	-130.156	-125.561	216.1265	-88.9582	-90.7558	162.2609	-171.606	-70.8071	-6.71261	-137.201	
U_2_St_Pe	-335.351	-455.79	263.8982	254.7088	333.6023	-200.068	278.9593	710.9489	349.8768	349.1924	-403.402	267.8098	392.0264	645.1821	291.2499	267.3018	253.914	264.2299	282.5766

Table 6.2 – An example of the results obtained using the Hyper-spheric distance measure with the original coin waveform data

6.4 *Wavelet Transform – Resampled Data*

This set of tests was conducted with new DWT coefficients calculated from the trimmed and resampled coin waveforms.

6.4.1 Hyper-spheres

Using the resampled data the results from the Hyper-spheric distance metric had improved but were still unsatisfactory. The best results attained are shown in Table 6.3.

Pos	Wavelet Used	Approx /Detail	Combination of Decomposition levels	Starting Decomposition Level	Minimum Distance between Hyper-spheres	Number of Clashing Hyper-spheres
1	BiorSplines 2.8	a	1	6	-41.8444	7
2	Daubechies 4	a	1	6	-27.8761	8
3	BiorSplines 6.8	a	1	6	-31.2678	8
4	Coiflets 3	a	1	6	-26.0493	9
5	Daubechies 7	a	1	6	-26.1602	9
6	Coiflets 4	a	1	6	-26.9451	9
7	Daubechies 4	a	2	6	-31.8838	9
8	BiorSplines 2.6	a	2	6	-50.3247	9
9	Symlets 4	a	1	6	-33.8451	10
10	BiorSplines 4.4	a	1	6	-38.1419	10

Table 6.3 – Top ten results from distance metric

From the table, BiorSplines 2.8 gave the best results with only seven Hyper-Spheres clashing. The coefficients formed from the approximation side of the Wavelet Transform gave the best results this is because the approximation side of the transform removes the high frequency, noisy parts of the data. This action can be seen in the decomposition example figures shown in section 6.2. This behaviour leaves coefficient sets which can be more easily grouped together within limits, which has the effect of delivering smaller limit ranges which have less chance of clashing. This can also be seen to be true by examining the decomposition levels that gave these results. They are all from decomposition levels 6 and above which means that a great deal of the noisy elements of the waveforms will have been removed.

The Haar Wavelet, which is the favourite Wavelet for implementation into a low cost processor environment, as discussed in chapter 3, gave a reasonable performance at position 72 with only 13 Hyper-spheres out of the possible 171 clashing. (See Table 6.4)

Pos	Wavelet Used	Approx /Detail	Combination of Decomposition levels	Starting Decomposition Level	Minimum Distance between Hyper-spheres	Number of Clashing Hyper-spheres
72	Haar	a	2	6	-45.0829	13

Table 6.4 - location of Haar Wavelet result

As the results show, a recognition solution is not possible with this distance metric, as a solution without clashing hyper-spheres does not exist. This result could be useful, however, if the number of coins within the coin set were to be reduced, particularly with the removal of the clashing coins. Then the distance metric would be able to separate the remaining coins. This means that this method could be utilised for a low quality coin recognition option, with a low number of coins in the recognition set and were high degrees of fraud protection are not important.

6.4.2 Hyper-cubes

This method was very successful in locating solutions that could separate all 18 coins in the test coin set. Out of the possible 2035 combinations of Wavelets and decomposition levels (37 filters * 55 decomposition combinations) 909 solutions were found.

Significantly, amongst this vast number of solutions were 12 configurations of decomposition level and filter using the Haar Wavelet. This is a superb result for implementation into a low power microprocessor solution.

When successful results had been located, it became necessary to analyse the results for a solution that was efficient in terms of the least number of coefficients required.

6.4.3 Coefficient significance

Table 6.5 shows selected results from the coefficient significance tests. The results displayed correspond to the minimum number of coefficients required for successful recognition to occur with each Wavelet type. Notably, with all the Wavelet Results the approximation coefficients provide this solution. The reason for this behaviour is the same as that found in the hyper-sphere results in section 6.4.1. Which is because the approximation results have had the noisy parts of the data removed the limits produced will be tighter and so will be found to clash less regularly.

An interesting result is that from BiorSplines 2.2 which failed to provide a solution which could separate all the coins. This can be explained by observing the shape of the Wavelet itself, which can be seen in Figure 6.6. The Wavelet is very spiky in nature and so would be effective at encoding the noisier aspects of the coin waveforms. However it would not correlate well with the slower curve shapes of the coin waveforms and consequently would not separate the coin data into different coefficients efficiently.

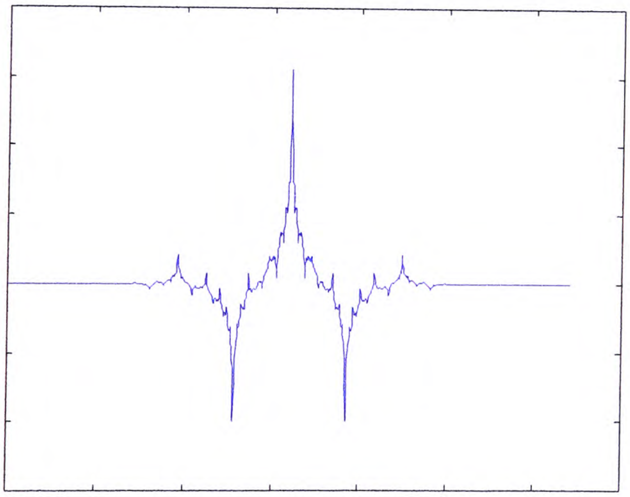


Figure 6.6 - BiorSplines 2.2

The Haar Wavelet performs well requiring only 10 coefficients to be used from decomposition levels 2,3,4,5. The Daubechies 9 also presents a good result requiring the least number of decomposition levels and coefficients recorded.

Filter	Approx / Detail	Starting Decomposition Level	Number of Combined Decomposition Levels	Number of Coefficients Required for Recognition
Haar	a	2	4	10
Daubechies 2	a	1	5	16
Daubechies 3	a	3	3	9
Daubechies 4	a	2	2	14
Daubechies 5	a	3	4	15
Daubechies 6	a	2	4	17
Daubechies 7	a	2	4	17
Daubechies 8	a	2	4	38
Daubechies 9	a	2	2	13
Daubechies 10	a	3	3	21
Coiflets 1	a	3	3	7
Coiflets 2	a	2	4	12
Coiflets 3	a	3	3	10
Coiflets 4	a	2	4	11
Coiflets 5	a	3	3	11
BiorSplines 1.1	a	2	4	10
BiorSplines 1.3	a	2	4	16
BiorSplines 1.5	a	2	4	22
BiorSplines 2.2	No Solution			
BiorSplines 2.4	a	1	5	16
BiorSplines 2.6	a	2	4	7
BiorSplines 2.8	a	2	4	20
BiorSplines 3.1	a	2	5	14
BiorSplines 3.3	a	3	3	12
BiorSplines 3.5	a	4	3	16
BiorSplines 3.7	a	4	3	37
BiorSplines 3.9	a	4	3	30
BiorSplines 4.4	a	1	5	17
BiorSplines 5.5	a	2	2	14
BiorSplines 6.8	a	2	4	17
Symlets 2	a	1	5	16
Symlets 3	a	3	3	9
Symlets 4	a	3	3	9
Symlets 5	a	3	3	7
Symlets 6	a	3	3	15
Symlets 7	a	4	2	21
Symlets 8	a	2	2	15

Table 6.5 – Minimum number of coefficients required for successful recognition for each filter type

6.5 WaveActiveX

After successful recognition combinations and significant coefficients had been located, the *WaveActiveX* program was configured to emulate the Wavelet coin recognition system with several of the configurations.

The configurations were then tested in two ways:

1. Passing all the coins in the recognition set through the mechanism.
2. Passing coins through the mechanism that were known to be causes of fraud with the original Tetrel Technology coin mechanism.

Examining Test 1 results:

The results from these tests were very positive, with all the configurations, using the least number of significant coefficients (listed in Table 6.5), 100% coin recognition was achieved. However, simpler Wavelet shapes such as the Haar were more sensitive to variations in the coins physical travel through the mechanism, which did result in recognition failures on occasional passes of the coins. Interestingly, throughout all the testing, coins which did not validate correctly did not result in incorrect identification either (i.e. incorrect validation would be a coin being identified as one of the other coins in the recognition set). This is a significant result as it would be most undesirable for coins to be incorrectly identified, especially in the case of a low value coin being validated as a high value coin.

Examining Test 2 results:

When the system was configured with the Haar Wavelet, it was found to provide more robust solution to fraud coins than the original Tetrel coin mechanism. However, coins that were almost identical to the coins in the recognition set, both dimensionally and in metallic composition, were validated correctly for approximately 50% of the fraudulent attempts. An example of a coin which demonstrate this kind of fraud is the Swedish 1 Krona which is very similar to the UK 10 Pence (see Figure 6.7).

Figure 6.7 – The Swedish 1 Krona (left) pictured with a UK 10 Pence(right)

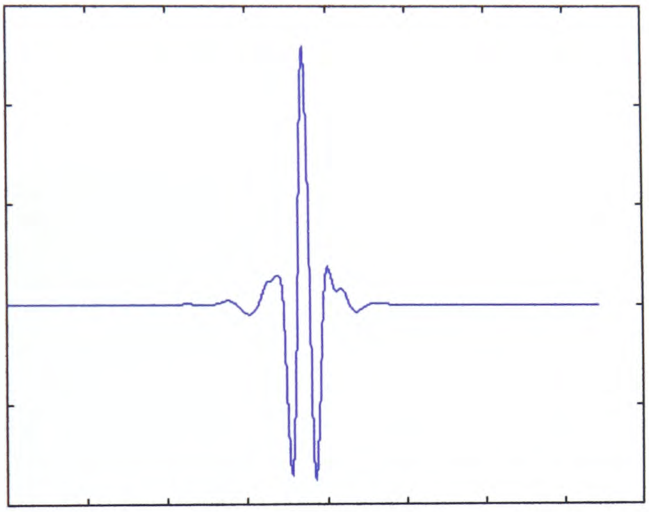


Figure 6.8 – The Coiflets 5 Wavelet

Wavelets that were of a more complex nature such as the Coiflets 5 (see Figure 6.7) proved to be extremely sensitive to the details in the coin waveforms. For Example, when the system was operating using this Wavelet, then the one Krona coin was correctly identified as being fraudulent in nature. Screen shots of the *WaveActiveX* program validating the UK 10 Pence and rejecting the Swedish 1 Krona can be seen in Figure 6.9 and Figure 6.10 respectively.

Interestingly, the Coiflets 5 Wavelet was so sensitive to coin details that it imbued the system with the ability to discern the orientation of the thicker coins in the coin set. For example, the orientation of the UK 1 pound coin could be ascertained as being heads uppermost or tails uppermost when passing through the mechanism.

After an examination of the results of the above analysis was made, it was agreed that the increased sensitivity of the systems was provided by the extended time localisation properties of the more complex Wavelets.

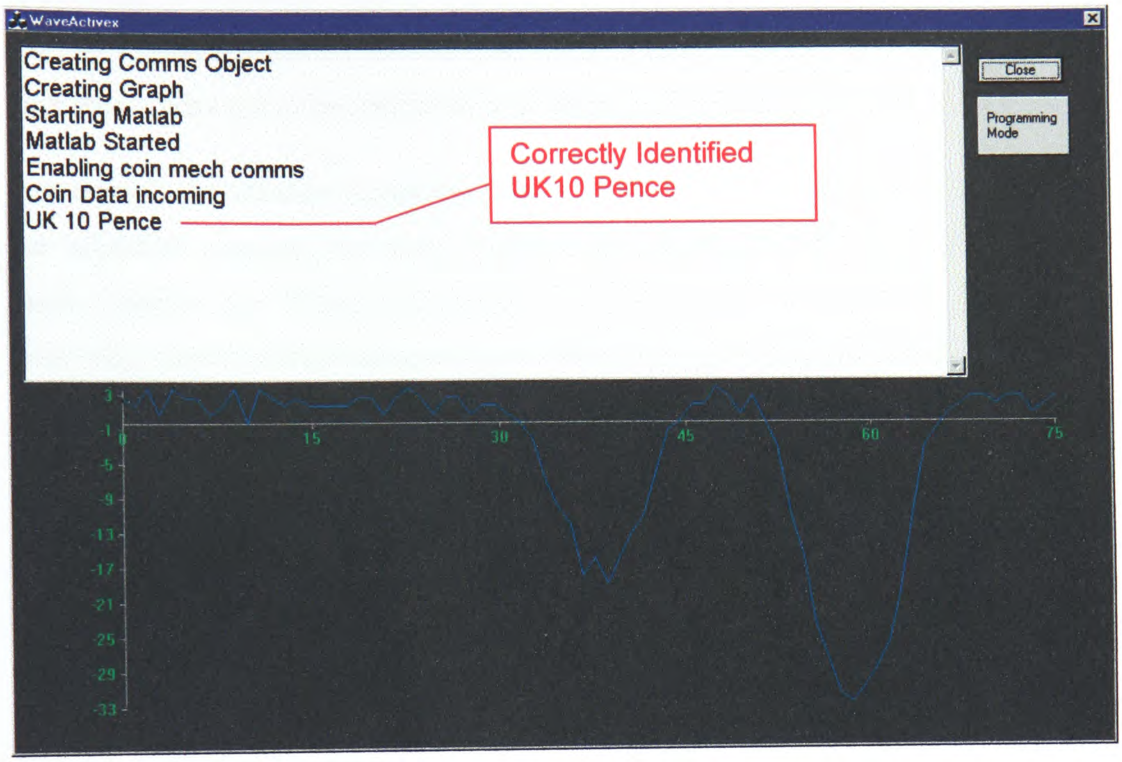


Figure 6.9 – WaveActiveX correctly Validating the UK 10 Pence

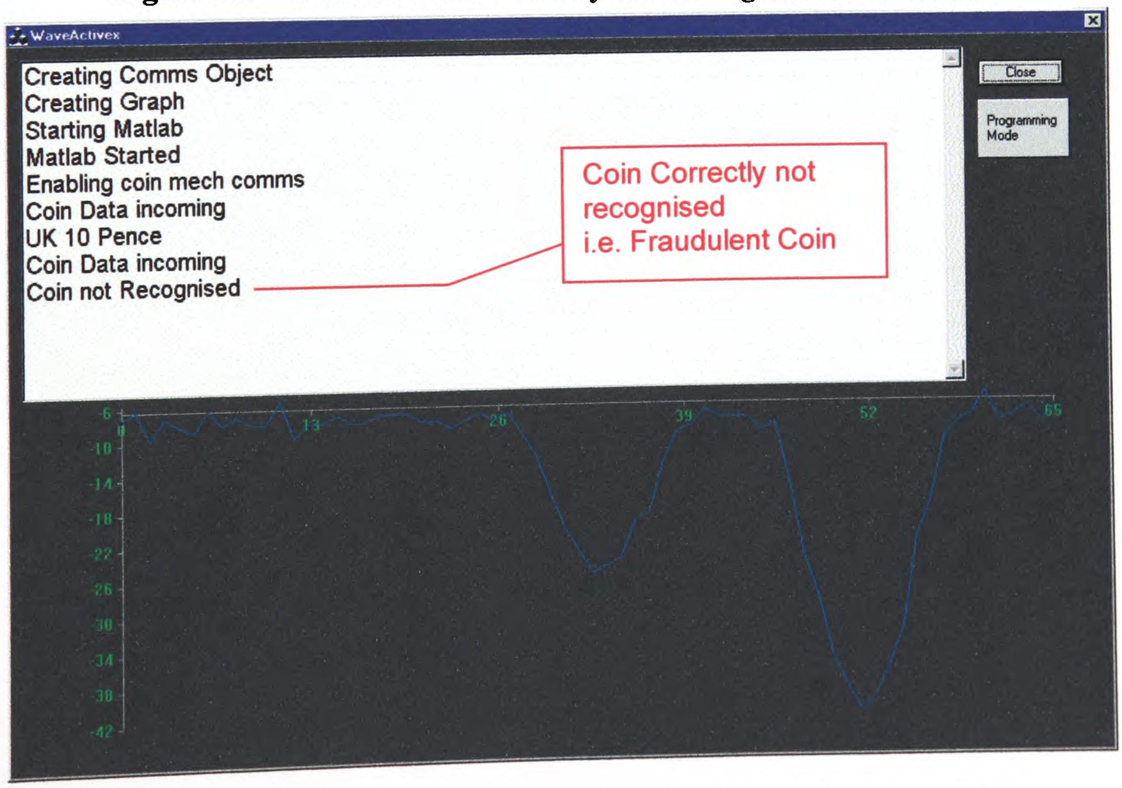


Figure 6.10 – *WaveActiveX* correctly not validating the Swedish 1 Krona

In summary of these tests, it can be stated that the results from Test 1 prove that the coin mechanism is a suitable replacement for the Tetrel coin mechanism and as a general-purpose coin validator. The results from Test 2 prove that the DSP mechanism is a distinct improvement on the original coin mechanism and that this kind of solution would be suitable for a task where fraud could be costly or where high value coins are to be controlled and validated with a high degree of accuracy.

Interestingly the solution is scalable, that is to say that the more complex and accurate the validation required, the more complex the Wavelet would have to be; and the more complex the Wavelet is, more processing power is required. When more processing power is required, a more powerful and expensive a processor is needed.

Therefore the solution can be scaled to the accuracy of validation required simply by increasing/decreasing the power of the processor together with increasing/decreasing the complexity of the Wavelet used.

Chapter 7 - Conclusions and Future Directions

This thesis has successfully addressed the application of Digital Signal Processing and Wavelet technologies to the field of automatic coin recognition.

Elementary DSP algorithms have been integrated into a low cost microprocessor based solution for coin recognition applications. This was developed into the low power TPP1 peak and trough location recognition coin mechanism.

Exploitation of time localisation properties of Wavelets for recognition has been proven. This was enabled by a resampling of input coin waveforms to allow for any temporal misalignment. The difficulties in locating an analytical solution were circumvented by the novel use of Data Mining combined with the use of a distance metric strategy.

The resampling, data mining and distance metric strategy were successfully converted into an integrated framework which can provide solutions for automatic coin recognition systems. This was proven by the development of a real time automatic coin recognition system emulation.

The new integrated framework enables improved recognition rate and lowers the potential for fraud dramatically within automatic coin recognition systems.

The framework also holds significant potential for future development, as it can be used to provide scalable solutions for integration into newer, more powerful, cost-effective processors.

The Resampling, Data Mining and distance metric strategy also hold potential for use in other recognition system, as they could find applications in any system which requires recognition of time warped, transient waveforms.

The recommendations for advancement of this research are as follows:

- Implementation of Wavelet algorithms in a low cost fixed-point processor - this will prove challenging because of truncation and rounding of numbers used in the calculations.

- Testing significant coefficients along with an estimation of processing effort required by each Wavelet to find the minimum processing requirement for successful recognition

- Development of unique Wavelet tailored specifically for the coin waveform - this may enable greater fraud protection and the ability to recognised a larger coin set.

- Experimentation with Neural network Multi-layer Perceptron device for processing outputs from DWT to obtain the best possible recognition rate i.e. to locate tacit solutions which are not available through standard distance metrics.

- Testing of the framework to recognise transient waveforms in other systems

Bibliography

1. Ali N. Akansu and Richard A. Haddad, Multiresolution Signal Decomposition, 1992

2. Atmel: Avr Application Notes, Atmel , version 1.2, 4th July 1997, file avr200.asm

3. Coin Validators, United States Patent number 5,687,829, Inventor: James Churchman, Filed: 8th January 1997 (see Appendix A for Patent Abstract)

4. Small Inductors, United Kingdom Patent pending, application reference 5232701, Applicant: Tetrel Limited, Filed: 19th November 1997 (see Appendix B for Application Abstract)

5. Charles K. Chui, Wavelets: A Tutorial in Theory and Applications, Academic Press, 1992

6. Ingrid Daubechies, Ten Lectures on Wavelets, 1992

7. Ingrid Daubechies, The Wavelet Transform time –frequency localisation and signal analysis, IEEE Trans. Inform. Theory, 1990

8. Ingrid Daubechies, Orthonormal bases of compactly supported Wavelets, 1988

9. Barbara Burke Hubbard, The World According to Wavelets, 1996

10. G Kaiser, A Friendly Guide to Wavelets, 1994

11. A. K. Louis, P. Maaβ and A.Rieder, Wavelets Theory and Applications, 1997

12. Mac A. Cody, The Wavelet Packet Transform, 1992

13. Mac A. Cody, The Fast Wavelet Transform, Dr Dobb's Journal April 1992

14. Mac A. Cody, A Wavelet Analyser, 1992

15. S. Mallat, "A theory for multiresolution signal decomposition: the Wavelet representation", IEEE Pattern Analysis and Machine Intelligence, vol. 11, no. 7, 1989

16. Matlab Wavelet Toolbox User Guide, 1998

17. Yves Meyer, Wavelets : Algorithms and Applications, 1993

18. Ronald Coifman, Yves Mayer, and Victor Wickerhauser, Wavelet Analysis and Signal Processing, 1991

19. Microsoft Developer Network Library, Visual Studio 6.0

20. Colm Mulcahy, Plotting and scheming with Wavelets, 1992

21. F.J.Owens, Signal Processing of Speech, 1994

22. John G. Proakis and Dimitris G. Manolakis, Digital Signal Processing Principles algorithms and Applications, Third edition, 1996

23. Wavelets and their Applications, edited by Mary Beth Ruskai, Jones and Bartlett Publishers, 1991

24. G. Streng, Wavelet Transforms verses Fourier Transforms, 1993

25. Christopher Torrence and Gilbert P. Compo, A Practical Guide to Wavelets Analysis, 1997

26. Victor Wickerhauser, Adapted Wavelet Analysis from theory to Software, 1994

27. B. Liu and S. F. Ling, "On the selection of informative Wavelets for machinery diagnosis", Academic Press, 1998

28. S. Mallat and Z. Zang, "Matching Pursuit With Time-Frequency Dictionaries", IEEE Transactions in Signal Processing, 1992

Appendices

Appendix A - Abstract – Coin Validators US Patent

US Patent number 5,687,829

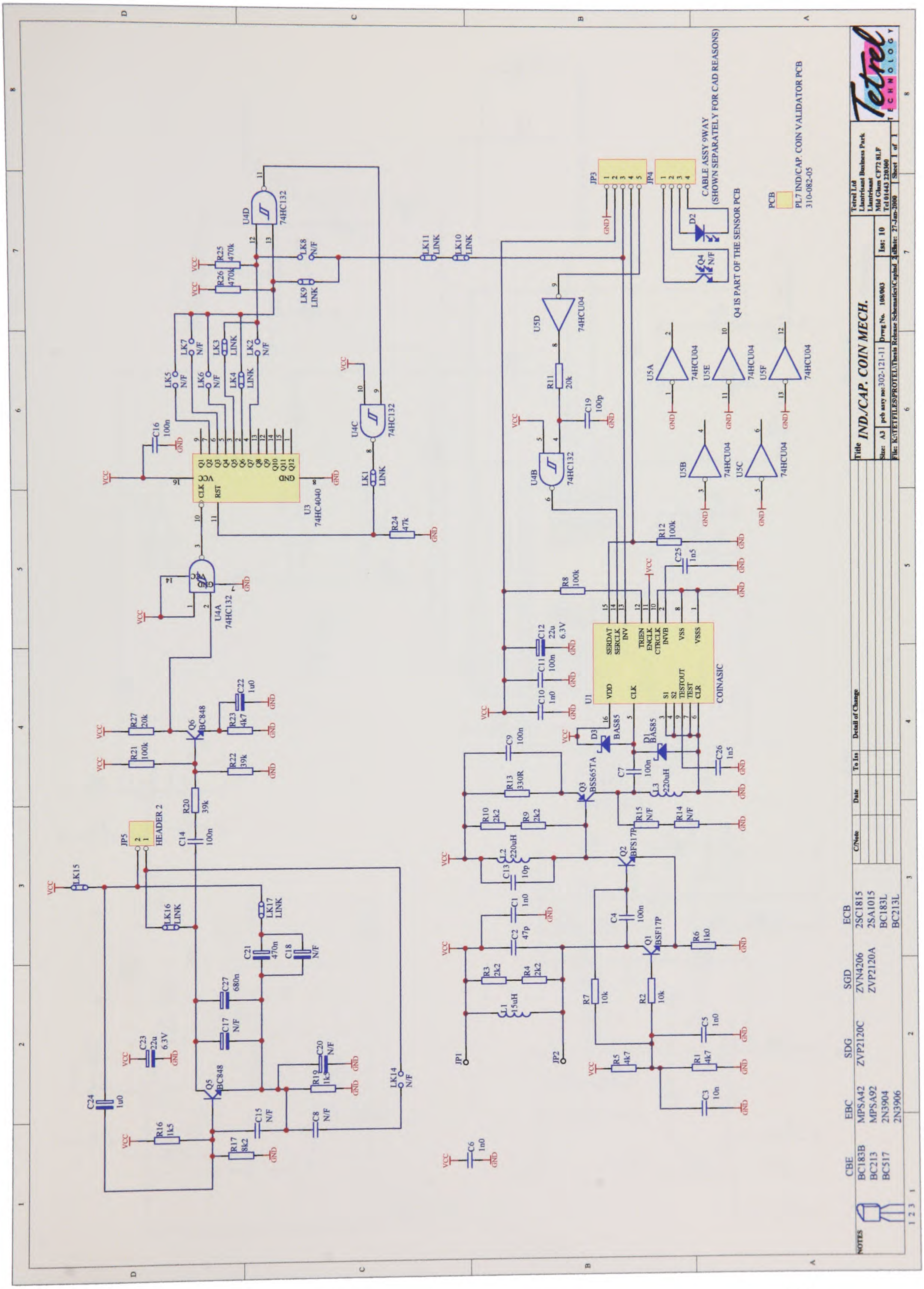
In a coin detection system, a coin passes between conductive plates which form a capacitor, which provides part of the frequency controlling capacitance of an LC tuned oscillator circuit. The presence of a coin between the conductive plates alters the capacitance, and consequently alters the output frequency of the oscillator circuit. The oscillator output is supplied to the clock of a counter, optionally following frequency division in a frequency divider. The counter counts the number of clock pulses received in a 10 ms period, and the count value is provided to a microprocessor via a shift register. The count value will be a measure of the frequency of the output of the oscillator. The microprocessor uses the output to determine whether a valid coin has been received, and if so, what the denomination of the valid coin is.

Appendix B - Abstract – Small Inductors UK Patent Application

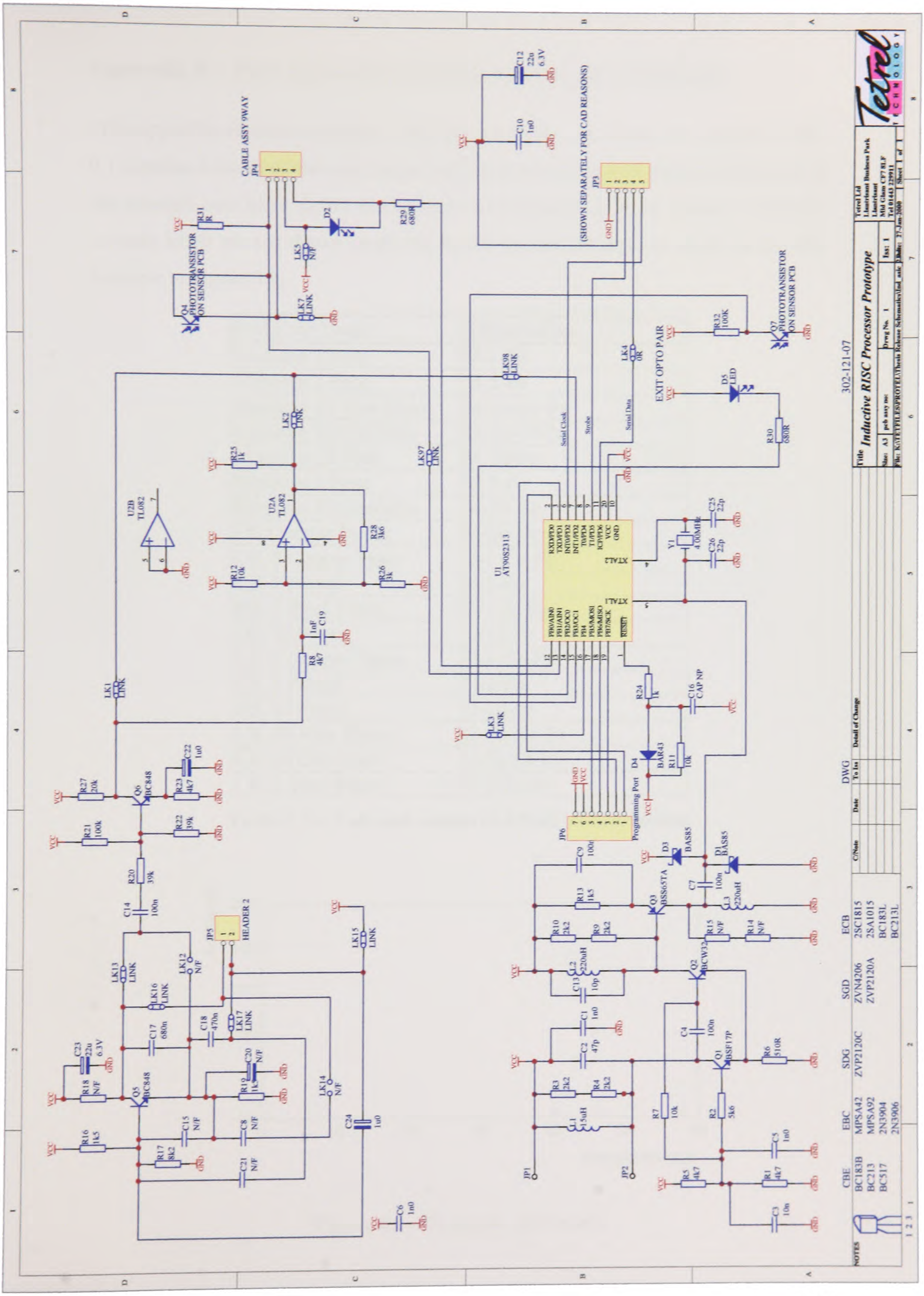
Patent Application number 5232701

A coin validation arrangement, usable for example in pay telephones, uses one or more inductive sensors having a small effective magnetic field so that the inductive sensor responds only to the material of a strip across the coin. Preferably a plurality of inductive sensors are used, mounted at different heights above the floor of the coin guide at different positions along the coin path. At each position along the coin path there may be either on or a plurality of inductive sensors. Such arrangements are particularly useful for recognising coins having an outer ring made of a different material from the central disc, and for distinguishing such coins from uniform composition coins.

Appendix C – Inductive and Capacative Coin Mechanism Circuit Diagram



Appendix D – Inductive RISC Processor Coin Mechanism Circuit Diagram



Appendix E – Coin Type Abbreviations and Sample Waveforms

This Appendix contains examples of all the coin types used during the research. Table 0.1 contains a list of all the coin names with their corresponding abbreviations used in the example coin plots shown in Figure 0.2 and Figure 0.3. In the interest of brevity the axis labels are not shown on all the figures but are the same as shown in the axis example in Figure 0.1.

Full Coin Name	Abbreviation
Mexican 1 Peso	M-1-Pe
Mexican 2 Peso	M-2-Pe
Mexican 10 New Peso	M-10-Ne-Pe
Mexican 10 Old Peso	M-10-Ol-Pe
Mexican 20 Peso	M-20-Pe
Mexican 5 Peso	M-5-Pe
Mexican 50 Centarvo	M-50-Ce
US Quarter Dollar	U-Qu-Do
UK 1 Copper Pence	U-1-Co-Pe
UK 1 Pound	U-1-Po
UK 1 Steel Pence	U-1-St-Pe
UK 10 Pence	U-10-Pe
UK 2 Copper Pence	U-2-Co-Pe
UK 2 Pound	U-2-Po
UK 20 Pence	U-20-Pe
UK 50 New Pence	U-50-Ne-Pe
UK 50 Old Pence	U-50-Ol-Pe
UK 2 Steel Pence	U-2-St-Pe

Table 0.1 – Full coin names and their abbreviations

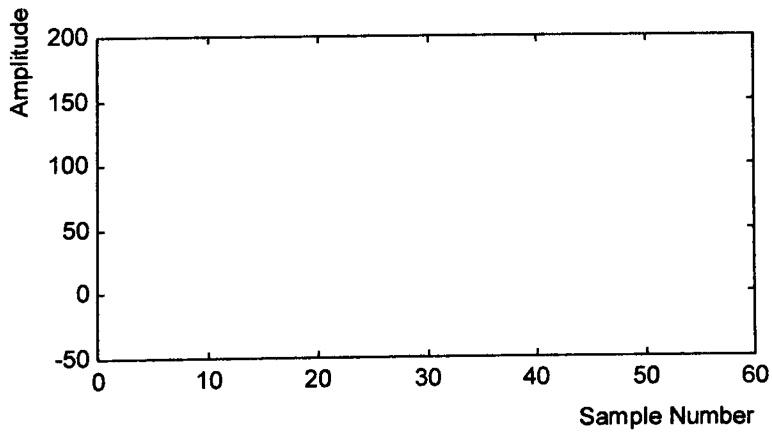


Figure 0.1 - Example Axis labels

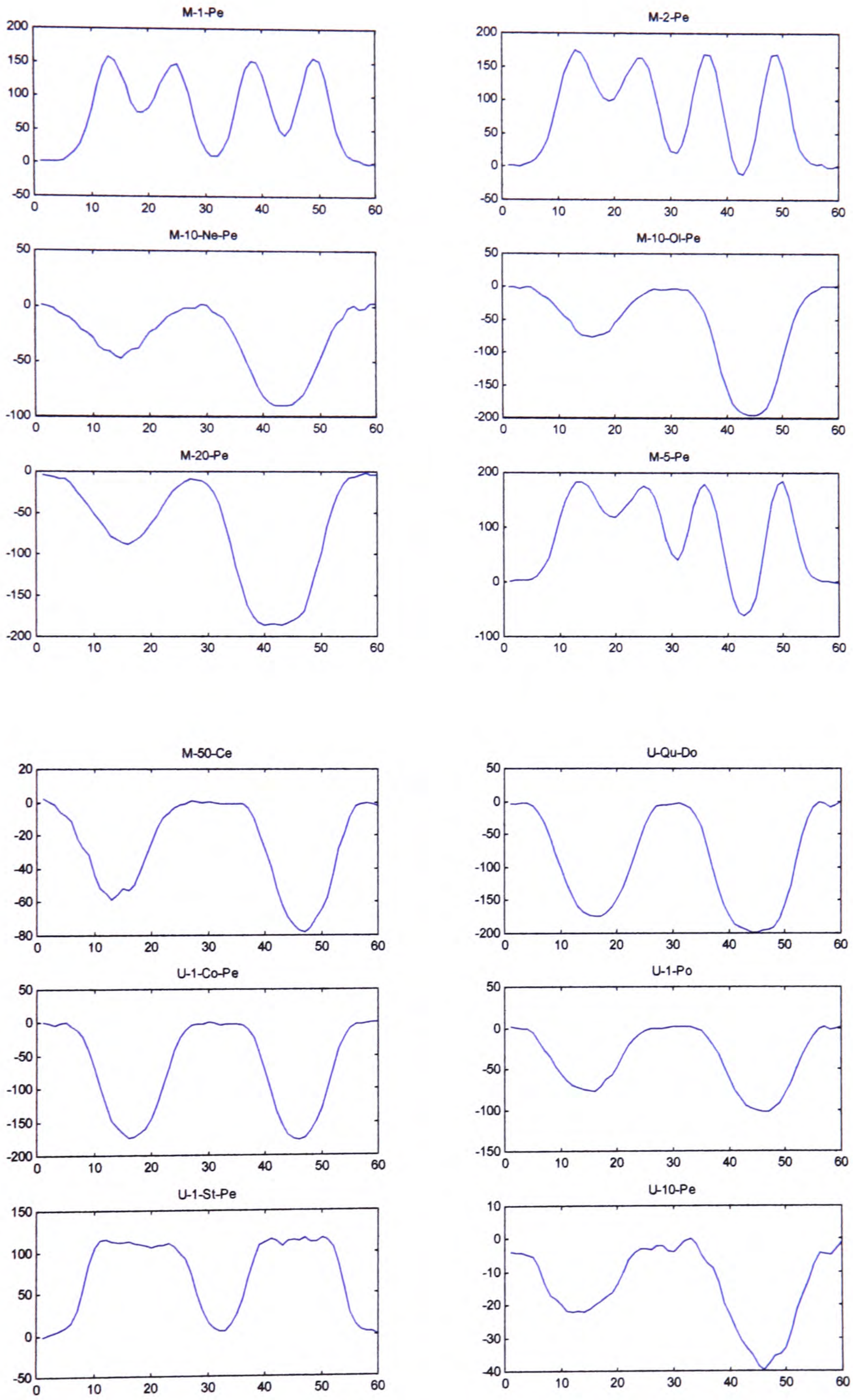


Figure 0.2 – example coin waveforms 1

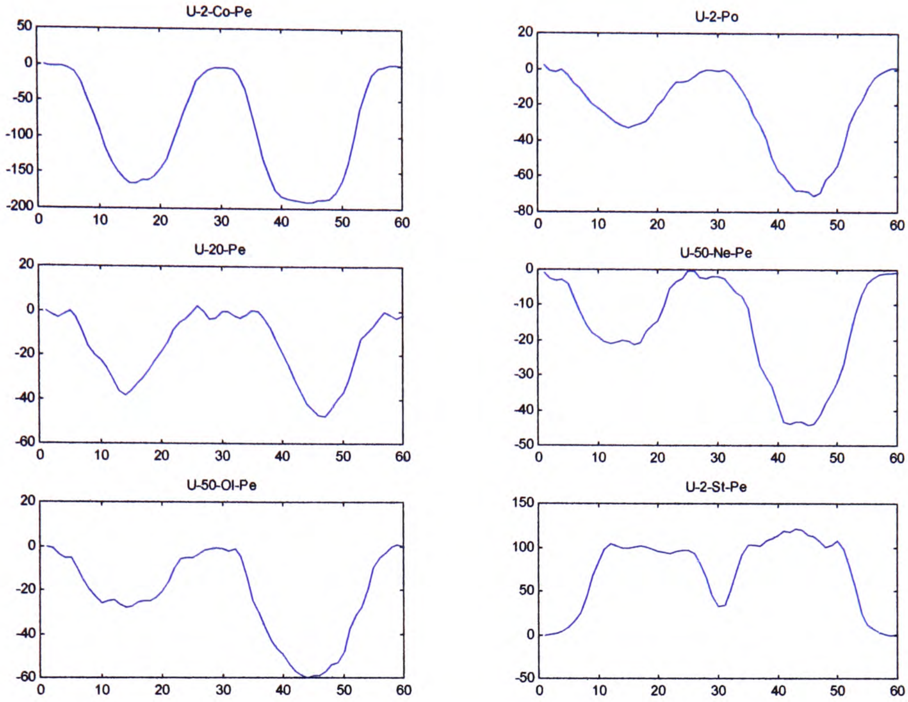


Figure 0.3 – example coin waveforms 2